Learning System of Web Navigation Patterns through Hypertext Probabilistic Grammars*

Sistema de Aprendizaje de Patrones de Navegación Web Mediante Gramáticas Probabilísticas de Hipertexto

DOI: http://dx.doi.org/10.17981/ingecuc.11.1.2015.07

 Research Article - Reception Date September 23, 2014 - Acceptance Date: December 15, 2014

Augusto Cortez Vasquez
Master of Computer and Information Sciences, Universidad Nacional Mayor de San Marcos. Lima (Perú). acortezv@unmsm.edu.pe

To reference this paper:

Abstract--- One issue of real interest in the area of web data mining is to capture users’ activities during connection and extract behavior patterns that help define their preferences in order to improve the design of future pages adapting websites interfaces to individual users. This research is intended to provide, first of all, a presentation of the methodological foundations of the use of probabilistic languages to identify relevant or most visited websites. Secondly, the web sessions are represented by graphs and probabilistic context-free grammars so that the sessions that have the highest probabilities are considered the most visited and most preferred, therefore, the most important in relation to a particular topic. It aims to develop a tool for processing web sessions obtained from a log server represented by probabilistic context-free grammars.

Keywords-- Probabilistic Grammars, Navigation Patterns, Pattern Learning Hypertext Probabilistic Grammar, Hypertext, Information Retrieval.

Resumen-- Uno de los problemas que reviste real interés en el área de minería de uso de la web es capturar las actividades de los usuarios durante su conexión y extraer patrones de comportamiento que permitan definir sus preferencias con el fin de mejorar el diseño de futuras páginas adaptando las interfaces de los sitios web a los usuarios individuales. En esta investigación se pretende ofrecer en primer lugar una presentación de los fundamentos metodológicos del uso de lenguajes probabilísticos para identificar sitios web más relevantes o visitados. En segundo lugar se representa las sesiones web mediante grafos y gramáticas libres de contexto probabilísticas de tal forma que las sesiones que tengan mayor probabilidad son consideradas las más visitadas o más preferidas, por tanto las más relevantes en relación a un tópico determinado. Se pretende desarrollar una herramienta para procesamiento de sesiones web obtenidas a partir de log de servidor representado mediante gramáticas probabilísticas libres de contexto.

Palabras claves-- Gramáticas probabilísticas, patrones de navegación, aprendizaje de patrones, gramática probabilística de hipertexto, hipertexto, recuperación de información.

* Research paper deriving from the research Project “Categorización de textos mediante máquinas de soporte vectorial.”. Funded by el Consejo Superior de Investigaciones UNMSM Lima – Perú. Starting date: January 2012. Ending date: December 2012.
I. INTRODUCTION

What science and technology have achieved so far has been truly spectacular. We just have to look around to witness what the extraordinary power of our understanding of nature has helped us achieve. In the early eighties the first text mining endeavors were made with the inconvenience of needing a lot of human effort, but technological advances have allowed this area surprisingly progress in the last decade. Text mining is a multidisciplinary area based on information retrieval, data mining, machine learning, statistical and computational linguistics. Like most of the information (over 80%) is currently stored as text, it is believed that text mining has great commercial value. When users browse the Web and want to retrieve pages in relation to a particular concept, they should avoid many irrelevant pages; the objective is therefore to recover significant pages, that is, those that are authority on the subject.

There are two related concepts: most relevant and most visited pages. Therefore, we start from the premise that the most relevant pages are those that are most visited. This research captures, from the information contained in the server logs, the users’ activities during their connection to the web and extracts behavioral patterns that will help understand the preferences of users’ browsing, allowing adapting the interfaces of future pages to individual users. To achieve this purpose, a simple model of hypertext represented by graphs was used; that is, a representation of the users’ navigation sessions which were inferred from the log files as a hypertext probabilistic grammar.

II. OBJECTIVES

A. General Objective: To obtain a tool to identify the preferences of users on the Web.

B. Specific Objectives:
1. To represent the web session by directed graphs.
2. To represent web sessions using hypertext probabilistic context-free grammars.

III. CONCEPTUAL FRAMEWORK

A. Information Retrieval

Information retrieval (IR) is a term used in a very broad sense that requires precision; it is often vaguely defined, and in this context refers only to automated information retrieval systems. Contreras points out in her thesis [1] that:

“Lancaster provides a definition: “An information retrieval system does not inform (i.e., change the knowledge of) the users on the subject of their inquiry. It merely informs on the existence (or non-existence) and whereabouts of documents relating to their request.”

The referred automated systems require speed, consistency, accuracy and ease of use in the retrieval of relevant texts to satisfy users’ queries.

B. Web Mining

There is a growing need to know how users interact with websites. Web mining (WM) essentially concerns with the discovery and analysis of users’ information on the web in order to uncover behavior patterns. Alcívar refers to the term WM as technology used to discover non-obvious information from data sources that include server logs [2].

C. Formal Language

Although a natural language is governed by grammatical rules that are already defined, they can be modified later (see Fig.1). This is an advantage for natural language, because this possibility enriches language, yet at the same time, it hinders its computer processing since it can be ambiguous and imprecise. On the contrary, a formal language is unambiguous and exact; it is a language developed by man to express situations that occur specifically in each area of scientific knowledge. Formal languages can be used to model a theory of mechanics, physics, mathematics, electrical engineering, or otherwise, with the advantage that in these languages all ambiguities are eliminated. Of particular importance are computer programming languages which are defined considering a set of lexical components, grammatical rules and semantic delimitation [3], [4].

1. Definition of Alphabet A: An alphabet A is defined as a finite set of symbols. The elements of an alphabet constitute the basic units or primitives of a language. These, in turn, are grouped into strings [5], [6]

2. Definition of Word: It is called string or word on an alphabet A, to a finite sequence of elements of A [7]

D. Grammar

A grammar G is a linguistic and mathematical model that describes the syntactic order to be met by well-formed sentences of a language [8], [9]. A grammar is formally defined as in (1):

\[ G = (V_T, V_N, P, S) \]  

Where:
VT: finite set of terminal symbols of language  
VN: finite set of non-terminal symbols  
P: finite set of production rules  
S ∈ VN: distinguished symbol or initial axiom  

From axiom S, sequences L are recognized by applying successively the rules on production grammar.

E. Probabilistic Context-Free Grammar

Chomsky classified grammars according to the form of its production rules, thus a context-free grammar has its rules as follows:

\[ P: A → a \]

Where:
\[ A ∈ V_N \quad Y a ∈ (V_N ∪ V_T) \]

The left side contains only a non-terminal, while the right side consists of a sequence of terminals and non-terminals [3], [8].

A probabilistic context-free grammar (PCFG) is a context-free grammar in which each rule is assigned a probability. The probability of a parsing is the product of the probabilities of each of the rules used in it. Thus there are analyses that are more consistent than others. Note that the PCFG extend the contexts-free grammars incorporating a probability function [2], [10].

A PCFG is then defined as fivefold \[ G = (V_T, V_N, P, S, £) \] where £ is a function to assign probabilities to each rule in P. Function £ expresses the probability that a non-terminal given will be expanded to sequence β. A probabilistic grammar has for each rule P a conditional probability.

\[ A → β \quad [p] \]

1. Assign Probabilities to Every Production Rule: After defining the grammar, a probability is assigned in each production rule (see Fig. 2)

Consider the following example taken from [3]

\[
\begin{align*}
P: & \{ \\
S → & \text{NAME VERB} \quad [1.0] \\
\text{NAME } → & \text{ADJ NAME} \quad [0.4] \\
\text{VERBOVER } → & \text{B } \text{→ SING ADVERB} \quad [1.0] \\
\text{ADJ } → & \text{he} \quad [0.25] \\
\text{ADJ } → & \text{she} \quad [0.25] \\
\text{ADJ } → & \text{the} \quad [0.15] \\
\text{ADJ } → & \text{those} \quad [0.15] \\
\text{ADJ } → & \text{small/naughty} \quad [0.10] \\
\text{VERB } → & \text{SING } \text{→ boy} \quad [0.50] \\
\text{VERB } → & \text{SING } \text{→ girl} \quad [0.50] \\
\text{VERB } → & \text{SING } \text{→ studies} \quad [0.27] \\
\text{VERB } → & \text{SING } \text{→ runs} \quad [0.16] \\
\text{VERB } → & \text{SING } \text{→ plays} \quad [0.34] \\
\text{VERB } → & \text{SING } \text{→ jumps} \quad [0.23] \\
\text{ADVERB } → & \text{fast} \quad [0.45] \\
\text{ADVERB } → & \text{slow} \quad [0.28] \\
\text{ADVERB } → & \text{much} \quad [0.27] \\
\end{align*}
\]

Fig. 2. Grammar with probabilities  
Source: [3]

The term hypertext refers to the organization system and presentation of data based on the linking of text fragments or graphics to other fragments, allowing the user to access information not necessarily sequentially but from any of several related items, as shown in the Fig. 3.

F. Hypertext Navigation

To understand more clearly the nature of navigation through the information hyperspace, it is necessary to decompose the problem as several authors have tried. In this sense, there is a discrepancy in the classification made by Wright and Lickorish, with the references [2] [11]; internal navigation, is what is part of the hypertext; and external, the one made possible by generic navigation tools, independent of hypertext. Hypertext navigation refers to the process of moving through multiple pages when you visit the Web.

G. Hypertext Probabilistic Grammar

A hypertext probabilistic grammar (HPG) is defined as \[ G = (V_T, V_N, P, S, £) \] and a regular grammar, defined by a regular expression, has a one-to-one relationship between \[ V_N \] and \[ V_T \].

Hernandez noted [2] that the sessions of users’ navigation inferred from the log files can be represented as a hypertext probabilistic grammar. Each non-terminal symbol belonging to \[ G \] corresponds to a visited page, each derivation rule corresponds to a link between pages. Thus, the rule \[ A → B \] means the transition from page \[ A \] to page \[ B \]. In this regard, it is important to note that this method consists of the fact that the strings generated by the grammar with the highest probability correspond to the users’ preferred paths [12].

The probability of a grammar string is the product of the probabilities of the productions used in its derivation [11].
Essentially server logs consist of one or more text files that are automatically created and managed by a server, where all activity that is done on it is stored. Each server, depending on its implementation and / or configuration may or may not create a particular log. One of the most typical logs is the access log of a web server that stores in each access and at the same time data such as an IP address, browser, date and time, etc., allowing the creation of the website statistics [2] [13].

IV. Methodology

The research was conducted with a sample of the server log files from the computer lab of the Systems Engineering Faculty. Using these files, a hypertext grammar (HG) was built; for this purpose, it was determined the number of times a particular grammatical rule was applied and statistical calculations were done by estimating the frequency in which the pages appear in the navigation session. For this purpose, each non-terminal symbol of HG corresponds to a page and each derivation rule to a transition from one page to another; then the probabilities of each of the production rules were assigned. To model the navigation sessions, a graph was constructed; and finally, a Java program was developed using the platform NetBeans IDE 7.3.

A. Grammar Definition

Grammar \( G \) was defined identifying the terminals, non-terminals symbols and derivation rules. A non-terminal symbol was assigned to each identified page.

B. Definition of Grammar \( HPG \)

The probability of each production rule associated to grammar is calculated.

C. Definition of Navigation Sessions

Using the server logs, a set \( P \) containing the navigation sessions was constructed.

D. Session Graph Construction

Sessions were modeled by a graph structure \( G \).

E. Implementation

A prototype was constructed to identify the most relevant pages.

V. Results

A. Definition of Hypertext Probabilistic Grammar

Using the navigation session set \( P \) obtained from the server log files, the identified pages were represented by non-terminals symbols of \( G \).

Production rules are displayed in Fig. 4, where the line is labeled with the probability \( P_{ij} \) resulting from derivation \( A_i \) to \( A_j \).

![Fig. 4. Transitions Diagram](Source: Author)

The next step was to perform statistical calculations to assign probabilities (see Table I). After determining the number of times pages were linked, it was calculated all middle and conditional probabilities and the number of times that a grammar rule has been applied.

<table>
<thead>
<tr>
<th>Rule</th>
<th>Occurrence of ( a )</th>
<th>Occurrence of ( a \rightarrow \beta )</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S \rightarrow A1A1 )</td>
<td>100</td>
<td>12</td>
<td>0.12</td>
</tr>
<tr>
<td>( S \rightarrow A2A2 )</td>
<td>100</td>
<td>3</td>
<td>0.03</td>
</tr>
<tr>
<td>( S \rightarrow A3A3 )</td>
<td>100</td>
<td>8</td>
<td>0.08</td>
</tr>
<tr>
<td>( S \rightarrow A4A4 )</td>
<td>100</td>
<td>9</td>
<td>0.09</td>
</tr>
<tr>
<td>( S \rightarrow A5A5 )</td>
<td>100</td>
<td>25</td>
<td>0.25</td>
</tr>
<tr>
<td>( S \rightarrow A6A6 )</td>
<td>100</td>
<td>33</td>
<td>0.33</td>
</tr>
<tr>
<td>( S \rightarrow A7A7 )</td>
<td>100</td>
<td>10</td>
<td>0.10</td>
</tr>
<tr>
<td>( A6 \rightarrow A2A7 )</td>
<td>50</td>
<td>16</td>
<td>0.32</td>
</tr>
<tr>
<td>( A6 \rightarrow A2A7 )</td>
<td>50</td>
<td>34</td>
<td>0.68</td>
</tr>
<tr>
<td>( A7 \rightarrow F )</td>
<td>15</td>
<td>15</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Source: Author

Then grammar \( G \) was expanded to a grammar \( HPG \). The productions are distinguished into two types:

1. \( Start \) Productions: those that begin with axiom \( (S) \) and represent the start of a session.
2. \( Transitive \) Productions: Those that start with a non-terminal different from \( S \) and correspond to the links between pages [2].

Table II shows the grammar with its probabilities:
LEARNING SYSTEM OF WEB NAVIGATION PATTERNS THROUGH HYPERTEXT PROBABILISTIC GRAMMARS

Table II. Grammar with Probabilities

<table>
<thead>
<tr>
<th>ID</th>
<th>Production Rule</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S → a1A1(0.12)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>S → a2A1(0.03)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>S → a3A3(0.08)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>S → a4A4(0.09)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>S → a5A5(0.25)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>S → a6A6(0.33)</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>S → a7A7(0.10)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>A1 → a2A3(0.35)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>A1 → a4A4(0.12)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>A1 → a3A7(0.23)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>A4 → a2A6(0.17)</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>A2 → a2A3(0.23)</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>A6 → a4A2(0.45)</td>
<td></td>
</tr>
</tbody>
</table>

Source: Author

Table III. Sessions of Navigation

<table>
<thead>
<tr>
<th>ID</th>
<th>Sesión</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A1 → A3 → A4 → A5 → A6 → A7</td>
</tr>
<tr>
<td>2</td>
<td>A1 → A7</td>
</tr>
<tr>
<td>3</td>
<td>A1 → A4 → A5 → A6</td>
</tr>
<tr>
<td>4</td>
<td>A1 → A1 → A6</td>
</tr>
<tr>
<td>5</td>
<td>A1 → A1 → A6 → A7</td>
</tr>
<tr>
<td>6</td>
<td>A1 → A1 → A6 → A7</td>
</tr>
<tr>
<td>7</td>
<td>A6 → A6 → A7</td>
</tr>
<tr>
<td>8</td>
<td>A6 → A7 → A7</td>
</tr>
<tr>
<td>9</td>
<td>A6 → A7 → A7 → A1</td>
</tr>
</tbody>
</table>

Source: Author

B. Sessions Graph

Production rules are shown in the following graph (see Fig 5), where the lines are labeled with the probability $P_{ij}$ resulting from derivation $A_i$ to $A_j$.

Where:

$S_i$: a session of set $P$

$A_i$: a page involved in a session $S_i$

$r_i$: the number of times a page $A_i$ was requested in the sessions $P$

$p_i$: the number of times a page $A_i$ was the first state in a session $S_i$ of $P$

$u_i$: the number of times a page $i$ was the last state in a session $S_i$ of $P$

$t_{ij}$: the number of times a subsequence of two pages appears on the session, or what is the same, the number of times the link was crossed of $P$

$a > 0$: strings can be generated from any state

$a = 0$: only states that took the top places in the current sessions have a probability higher than zero to be start production

$a = 1$: the probability of a start production is proportional to the number of times the corresponding state was visited. The destiny node of a production with higher probability corresponds to the state that was visited more often

$N$: $N ≥ 1$ determines the user’s memory when navigating the Web, that is, the number of previous URLs that may influence the choice of the following URL

If $N = 1$, the result will be what is formally known as a Markov string, which is a special type of discrete stochastic process in which the probability of an event occurring depends on the immediately preceding event. This lack of memory feature is called Markov property as shown in (2) and solved in (3):

$P(S \rightarrow a_1A_1) = \frac{\alpha \cdot N-V-A_1}{N-T \cdot V} + \frac{\alpha \cdot N-I-A_1}{N-T \cdot I}$

Where:

$N-V-A_1$: number of visits to $A_1 = 6$

$N-S-A_1$: number of starts from $A_1 = 4$

$T-N-V$: total number of visits $= 36$

$T-N-S$: total number of starts $= 8$

$P(S \rightarrow a_1A_1) = \frac{0.5 \times 6}{36} + \frac{0.5 \times 4}{8} = 0.33$
Using axiom $S$, symbols between $A_1$ and $A_7$ can be chosen. Applying the formula, it yields that page $A_1$ has higher probabilities to be selected, followed by $A_3$, $A_4$, $A_5$, and $A_6$; $A_2$ and $A_7$ are equally probable (Table IV).

**Table IV. Production Choice Statistics From Axiom S**

<table>
<thead>
<tr>
<th>Production</th>
<th>$p$</th>
<th>$\alpha$</th>
<th>NVA</th>
<th>NTA</th>
<th>NIA</th>
<th>NTI</th>
<th>P($p$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S \rightarrow a_1A_1$</td>
<td>0.5</td>
<td>6</td>
<td>36</td>
<td>4</td>
<td>8</td>
<td>0.33333333</td>
<td></td>
</tr>
<tr>
<td>$S \rightarrow a_2A_2$</td>
<td>0.5</td>
<td>2</td>
<td>36</td>
<td>0</td>
<td>8</td>
<td>0.02777778</td>
<td></td>
</tr>
<tr>
<td>$S \rightarrow a_3A_3$</td>
<td>0.5</td>
<td>4</td>
<td>36</td>
<td>2</td>
<td>8</td>
<td>0.18055556</td>
<td></td>
</tr>
<tr>
<td>$S \rightarrow a_4A_4$</td>
<td>0.5</td>
<td>7</td>
<td>36</td>
<td>1</td>
<td>8</td>
<td>0.15972222</td>
<td></td>
</tr>
<tr>
<td>$S \rightarrow a_5A_5$</td>
<td>0.5</td>
<td>6</td>
<td>36</td>
<td>1</td>
<td>8</td>
<td>0.14583333</td>
<td></td>
</tr>
<tr>
<td>$S \rightarrow a_6A_6$</td>
<td>0.5</td>
<td>6</td>
<td>36</td>
<td>0</td>
<td>8</td>
<td>0.08333333</td>
<td></td>
</tr>
<tr>
<td>$S \rightarrow a_7A_7$</td>
<td>0.5</td>
<td>2</td>
<td>36</td>
<td>0</td>
<td>8</td>
<td>0.02777778</td>
<td></td>
</tr>
</tbody>
</table>

Source: Author

This probability is shown in Fig 6

![Fig. 6. Comparative table of selected page probability from axiom S. Source: Author](image)

**VI. Conclusions**

This research emphasizes the importance of context-free grammars (widely used in language theory) as a tool to detect the preferences of website users. This instrument allows commercial companies to improve their websites to maximize the business impact in terms of the dynamic behavior of its visitors.

The method allowed inferring, from the log files, users’ navigation sessions representing them through hypertext probabilistic grammar, so that the sequences generated or recognized by the grammar correspond to preferred users’ sessions or paths.

The main difficulties of building probabilistic context free grammar were, first, to build the grammar, and then assign the probabilities in each production rule.

The developed model can be used to calculate the probability of reaching a page if the user is on a given page.

There are many tools for websites analysis and statistics that together with web servers provide really good data views and summaries to generate reports and graphs, but do not allow other activities like drawing patterns on user behavior or explore the relevance and ranking of pages. Our analysis of web sessions modeled by context-free grammars is equated with the ability to extract and use information from sessions to learn users’ behavior patterns. The patterns obtained from past uses can determine web customizing, meaning by customization any action that adapts the Web to suit the user.

Computational linguistics is not only a method but a paradigm with a computational scheme of language processing that has led to a wide variety of applications, in this case, to the learning of navigation patterns.

**References**


