The Design of Gugubarra 2.0: A Tool for Building and Managing Profiles of Web Users

Natascha Hoebel, Sascha Kaufmann, Karsten Tolle, Roberto V. Zicari
Databases and Information Systems (DBIS)
Johann Wolfgang Goethe-University of Frankfurt
Robert-Mayer-Strasse 11-15, D-60325 Frankfurt, Germany
{hoebel, kaufmann, tolle}@dbis.cs.uni-frankfurt.de
zicari@informatik.uni-frankfurt.de

Abstract

In [6] we have introduced the concept of non-obvious user profiles (NOPs) to capture the hypothetical interest of web users. In this paper we present the design principles and rules of our Gugubarra engine, which is a tool to calculate and visualize these non-obvious user profiles.

1. Introduction

With our *Gugubarra project* (Gugubarra is the Aboriginal name for the Kookaburra bird) we try to build user profiles representing the users' interests. User profiles can be used for recommender systems, which is not our main objective. Goals we have in mind are as follows:

- clustering of user groups by interests,
- detect shifts in interests of users or user groups,
- detect upcoming trends of the community.

Reaching these goals would enable the owner of a web site to understand the community that is visiting his site. He therefore could concentrate for example his efforts on those areas that are potentially of interest to the community. He could also get a feedback if his work is adopted as he anticipated.

The Gugubarra project began in 2004 within the database group (DBIS) at the Computer Science Institute of the Johann Wolfgang Goethe University, with the aim to build tools for better management of communities of registered Web visitors.

A first research prototype system, called Gugubarra 1.0 has been implemented in 2004, which allows to build and manipulate non-obvious user profiles [6]. It was showcased at the CeBIT Trade Fairs in 2005 and 2006 [1]. Gugubarra 1.0 works as a test application on real data provided by the Web community viewzone.org.

We are currently designing the second prototype called Gugubarra 2.0, where we offer various settings that can be used to influence the calculation of the profiles. By using these settings the Web site owner can focus on aspects he wants to analyze and adjust the tool according to his community or target group.

The design principle of the Gugubarra engine is as follows: NOPs are calculated by using several different parameters that can be chosen by the owner of a Web portal. The combination of the chosen parameters implements a specific *strategy* to deploy and manage NOPs. The common element for all strategies that can be chosen is as follows: The Creation of NOPs is done by looking at the behavior of the web user and by taking into account a feedback mechanism. More precise the approach we have in Gugubarra is as follows:

i. For each registered Web visitor we create a profile. These user profiles reflect the "inferred" interests of the users related to a set of pre-defined topics defined by the owner of the Web site. The profiles go beyond collecting the obvious information the user is willing to give at the time of registration. In Gugubarra, a user profile contains two parts: the obvious profile, given directly by the user and a non obvious profile (NOP), inferred by the user's behavior during his visits on the site.

ii. The user profile is (re)-calculated dynamically every time an explicit feedback is given by the user and/or a set of events occurred which are related to the user's behavior and to certain "locations" of the Web site.

iii. We cluster Web visitors by clustering similar profiles of interest [4]. Cluster of Web visitors can then be used to analyze patterns of interests in the Web community and to forecast further behavior. Clusters might also provide useful information to support the decision what kind of new E-services to introduce for the Web community and when to introduce them.

In this paper we'll give an overview of the differences between Gugubarra 1.0 and the new version. The effect of the various parameters, in defining a strategy to create the NOPs, will also be emphasized.

2. Gugubarra 1.0 Review

The approach to calculate the NOPs with Gugubarra 1.0 was only page and duration based. The strategy has been defined detailed in [6]. Basically it consists of associating each page *Pj* of the web site to a set of pre-defined topics $Tp_1,..,Tp_n$, which are weighted statically. A weight $v_j(Tp_i)$, $(0 \le v_j(Tp_i) \le 1)$ is defined for each page, representing the strength of the relation between the page P_j and the topic Tp_i.

Each time a user looks at a page P_j, the corresponding set of topics is considered depending on the associated weights. The initial NOP is then created by setting a value x_i, between 0 (*no interest*) to 1 (*strong interest*) to the associated topic list by considering the pages the user has looked at during a session. The values x_i of the NOP are calculated using the following formula, where $duration(P_j)$ takes into account the time spent by the user on the specific page (see Formula 1).

$$x_i = \frac{\sum_{j=1}^{m} duration(P_j) * v_j(Tp_i)}{\sum_{k=1}^{m} duration(P_k)} \qquad (1)$$

We discovered some issues that limited the approach of Gugubarra 1.0. First of all it is page based. One page can contain various parts with different content. Normally you do not know which part a user views inside his browser. However, in case he presses a link or enters some information, e.g. into a form, you could take these action of the user into account.

This should be reflected within the calculation. In addition many sites are generated by content management systems that build up their pages out of different smaller parts. As consequence we need to break down the pages to smaller units, which we will call zones.

In Version 1.0 only the duration is used. In addition the actions of the users should also be taken into account. Note that the duration is somehow tricky since you will never know what exactly the user is doing in front of the screen, if he stays there at all. In the next section we introduce zones, actions and other concepts we use with Gugubarra 2.0.

3. Concepts for the NOP calculation in Gugubarra 2.0

We consider a Web site (or a domain) as a collection of web pages that are linked together within the site. Each page has specific content.

In Gugubarra 2.0 the following main concepts are introduced to create a user's NOP: ***Zones*** with a state**,** ***Topics, Weights, Actions*** and ***Feedback***. There are several different parameters in Gugubarra 2.0 that combined determine a specific strategy to deploy and manage NOPs and influence the usage of the concepts. Some of these possible settings are described within the next section. Here we give a brief description of the main concepts.

A ***zone*** defines an area on the Web site. It can be a set of pages, a set of parts within a page, a set of parts of several pages, or any combination thereof.

A zone has one of three ***states***. The state *ON* indicates that this zone is being used to calculate the NOPs of the visitors, state *OFF* indicates the zone is not used. In the third state *OFF-ACTION-SENSITIVE*, only actions the user does within the zone are taken into account (formula (3)), not the duration (formula (4)), which will be described in the next section.

Topics are related to the content and are defined global by the owner of the Web site and then associated to the *zones*. A ***weight*** indicates the relative importance of the topic in respect to a scale from 0 (not relevant) to 1 (extremely relevant) in the zone.

Actions are also global defined by the owner of the Web site, so they are applicable to any zone defined for the Web site. Each action has an associated weight which indicates the importance given to such action by the owner of the Web site, ranging from 0 as minimum up to *n* as max.

The ***feedback*** is an integral part of the NOP approach. This mechanism is triggert due to certain criteria, where upon the user is requested to give his feedback. Here the user explicitly answers questions to determine his interests. We use the given feedback to measure the accuracy of the calculated NOP. However, we can not assume that all users will give correct answers, e.g., on purpose, due to lack of time or because of a misunderstanding. The influence on the NOP of the user is therefore limited by certain rules [3][4][6].

By the way, we believe that it is important that the user is able to see and determine his profile. We know that there are some difficulties relating to this approach, e.g. the possibility of the user to influence his profile. However, we think this is important for the user to feel comfortable and to trust the system.

4. Calculating the NOP

In this section we describe the general way of calculating the NOP.

In general the NOP is a set of values representing the level of interest for a set of topics. The calculations of these levels of interest (see formula (2) below) are determined by two parts:

i. ***Action Profile***: the actions a user does in the zones on the topic Tp_i, which is computed in the *Action Profile ActP(i)* and

ii. ***Duration Profile***: the time, a user spends on pages associated to the topic Tp_i, which is computed in the *Duration Profile DurP(i)*.

These two parts can be parameterized by the Web site owner in accordance to his needs. This means the owner can increase or decrease the impact of the time a user spends on a page compared to the actions he did.

It is worth mentioning that there are some problems related to the time a user spends on the page, because you will not know if the user is really reading this page or has fallen asleep. Of course this can partially be absorbed by using timeouts. However, by allowing the Web site owner to adjust the parameters to the measured behavior of the community, we expect better results and a higher level of trust the Web site owner gives to the results.

In the overall formula (2), the parameters *a* and *b* are used to customize the ratio between *ActP(i)* and *DurP(i)* (implemented by a slider in Gugubarra 2.0).

$$x_i = a * ActP(i) + b * DurP(i), \quad where\ (a+b) = 1 \qquad (2)$$

Let's have a closer look at the two parts of the formula (2), as defined by (3) and (4) below.

To compute *ActP(i)*, we determine zone *q*, where the action occurred and obtain the associated topic weight $v(Tp_i, Z_q)$. We then multiply this value by the sum of all weights for all occurred actions in this zone. Finally we calculate the sum of all zones, where an action occurred and the associated topic lists contain topic *i*, divided by the sum off all occurred action weights.

DurP(i) is computed in a similar way. We consider each visited page P_j, that contains the topic Tp_i and multiply the time the visitor spent on this page *duration(Pj)* by its topic weight $v(Tp_i, P_j)$. We finally sum these values and divide it by the total time.

$$ActP(i) = \frac{\sum_q \left(\sum_t aw_t * v(Tp_i, Z_q) \right)}{\sum_s aw_s} \qquad (3)$$

$$DurP(i) = \frac{\sum_j \left(duration(P_j) * v(Tp_i, P_j) \right)}{\sum_k duration(P_k)} \qquad (4)$$

In fact, it isn't easy to define the *duration*(P_j) used in formula (4). What we want is the time the user viewed the page. The problem is that from the log files we only know when he requested the page and when his next request was performed, in case there is a next request.

The Web site owner can define time limits that should be taken into account for the duration profile part. We did not include them into the formula (3) to keep it easier. However, the basic idea is simple. The Web site owner can define a minimum and maximum time for *duration*(P_j). Everything out of this range will not be considered.

In case a visitor visits a page less time than the specified minimum, the system expects the requested page did not contain the content the visitor expected. Therefore the duration part will not be taken into account. In case the visitor stays longer on the page than the defined maximum time, the system limits the influence of it, due to the fact that we do not know if the user is really engrossed or if he has fallen asleep.

As an additional fact we take the design of the Web site into account. The designer can decide whether the link shall be opened inside the same window overwriting the *parent page* or inside a new window. The later case is normally done by the designer to allow the user to perform additional actions on the more important parent page. Therefore we keep on counting the duration for parent page up to a defined value (by default the maximum value). In case Gugubarra 2.0 realizes that another action has been performed on the parent page the duration is further increased. Note: We only take the design view into account, of course the user is free to open links in new windows.

For the minimum and maximum time we are currently also working on an approach to take the amount of data contained inside the page into account. Of course for a page with very little content the minimum and maximum time should be smaller than for a Web page displaying a specification with more than 50 printed pages. Also the average time a specific user needs to read a word on a page could give an indication how to set the minimum and maximum time (*average reading speed*, see [5]).

Note that in formula (4) we used the value $v(Tp_i, P_j)$ to compute the duration part *DurP(i)*. This should be the weight of the topic Tp_i within the page P_j.

In *Gugubarra 2.0* however the owner associates topics and their weights to zones and not to pages. To calculate the NOP therefore we generate a *page topic list* composed by the *zone topic lists* for each page. For this calculation only zones with *state=ON* are taken into account. In addition the Web site owner can decide individually for

every page how this topic list is generated by using one of the following rules:

- MAX-Rule: For each topic Tp_i, scan all zones on a page *j* and take the maximum topic weight.
  $v(Tp_i, P_j) = \max(v(Tp_i, Z_q)) \forall Z_q \in P_j$
- MIN-Rule: For each topic Tp_i, scan all zones on a page *j* and take the minimum topic weight.
  $v(Tp_i, P_j) = \min(v(Tp_i, Z_q)) \forall Z_q \in P_j$
- AVG-Rule: For each topic Tp_i, scan all zones on a page *j* and take the average topic weight.
  $v(Tp_i, P_j) = avg(v(Tp_i, Z_q)) \forall Z_q \in P_j$

with Z_q is a zone with *state=ON*.

5. Choosing the Parameters to define a Strategy

Each community is different and behaves differently. It is therefore important for the Web site owner to be able to configure how the calculation of the NOPs for his community will be performed and what will be taken into account under which circumstances. For our next prototype system *Gugubarra 2.0* we therefore will give users of the system the ability to define parameters that will influence the NOP generation. Let's revise the effects of the parameter settings, done by the owner:

- Setting the topics and their weights for each zone. This is fundamental and will affect both parts of the formula (1).
- Define specific actions, the Web site visitors could perform, together with their relevant weights. This will influence the action profile part, represented in formula (2).
- Turning the states of the zones *ON* or *OFF*. This will influence the page topic list generation and therefore influence the duration profile part represented in formula (3).
- Using the different rules to calculate the page topic list (*MAX-Rule*, *MIN-Rule* or *AVG-Rule*) will influence the duration profile part.
- By specifying the factors *a* and *b* in formula (1), the action and duration profile parts can be balanced by the Web site owner. With the extreme case of *a=0* and *b=1* actions are turned off in the sense that formula (2) will not be considered for the NOP generation. However, actions still might have an influence by turning *ON* or *OFF* the state of zones and therefore can influence the page topic list generation.
- Defining the maximum and minimum duration that will be considered for the duration part.

A peculiarity of Gugubarra is the combination of NOPs, user feedback and custom rules for the generation of profiles. This is missing in systems such as [10][11]. Compared to systems such as Web Usage Mining [7] we included more granular information by the introduction of zones and actions, and we do not only rely on the recorded click stream like in [2], [8], [10]. A detailed example of the use of Gugubarra 2.0 can be found in [3].

6. References

[1] DBIS J.W. Goethe University, CeBIT Presentations for Gugubarra 1.0., http://www.dbis.informatik.uni-frankfurt.de/news/?mode=art&l=e&aid=2&tmid=3&smid=7, 2006.

[2] Y. Fu, K. Sandhu, and M. Shih, *Fast Clustering of Web Users Based on Navigation Patterns*, in World Multiconference on Systemics, Cybernetics and Informatics (SCI/ISAS'99), pages 560--567, Orlando, FL, 1999.

[3] Natascha Hoebel, Sascha Kaufmann, Karsten Tolle and Roberto Zicari, *The Gugubarra Project: Building and Evaluating User Profiles for Vistors of Web Sites,* First IEEE Workshop on Hot Topics in Web Systems and Technologies, Boston, Massachusetts, USA, November 2006.

[4] Natascha Hoebel and Roberto Zicari, *On Clustering Visitors of a Web Site by Behavior and Interests*, DBIS report, May 2006.

[5] T.-P. Liang and H.-J. Lai, *Discovering User Interests from Web Browsing Behavior: An Application to Internet News Services*, IEEE Computer Society, Los Alamitos, CA, USA, 2002.

[6] Naveed Mushtaq, Karsten Tolle, Peter Werner and Roberto Zicari, *Building and Evaluating Non-Obvious User Profiles for Visitors of Web Sites*, IEEE Conference on E-Commerce Technology (CEC 04), San Diego, California, USA, 2004.

[7] J. Srivastava, R. Cooley, M. Deshpande, Pang-Ning Tan, *Web usage mining: discovery and applications of usage patterns from Web data*, ACM Press, New York, USA, 2000.

[8] Qing Wang, Dwight J. Makaroff, H. Keith Edwards, *Characterizing Customer Groups for an E-Commerce Website,* Proceedings of the 5th ACM Conference on Electronic Commerce (EC '04). ACM Press, New York, 2004.

[9] Weinan Wang, Osmar R. Zaïane, *Clustering Web Sessions by Sequence Alignment*, Proceedings of the 13th International Workshop on Database and Expert Systems Applications, 2002.

[10] S. Acharyya, J. Ghosh*, Context-Sensitive Modeling of Web-Surfing Behaviour Using Concept Trees*, WEBKDD, 2003.

[11] B. D'Ambrosio, E. Altendorf, J. Jorgensen, *Probabilistic Relational Models of On-line User Behavior*, WEBKDD 2003, 2003.