Recognition of Internet Portal Users on the Basis of Their Behaviour

Wojciech Jaworski

Institute of Informatics
University of Warsaw, Banacha 2, 02-097 Warsaw, Poland
wjaworski@mimuw.edu.pl

Abstract. Our aim is to develop methodology for recognition of Internet portal users on the basis of their behaviours. This is a classification task in which we have thousands values of decision attribute and objects are described by means of sequences of symbols. We develop feature selectors which make our task tractable. Since the behaviour usually does not distinguish users, we introduce user profiles which are clusters of indiscernible users and we construct classifiers which assign descriptions of user behaviour with user profiles. We also derive specific for our task quality measures.

Keywords: Web traffic analysis, sequential data classification, quality measures.

1 Introduction

Web traffic analysis is an area of still growing importance. The ability of determining which pages were displayed by the same user is crucial for this domain. Usually users are identified by the cookies technology. However, this method fails in many cases: user may delete cookie, use more than one computer etc. The only trace that allow us to identify user in such case is his behaviour - the similarity between sequences of Internet pages visited by user which we try to recognize with behaviour of known users.

In this paper we analyze it in the context of the following case study: we are given an Internet portal that provides web pages. Each of them belongs to one of 20 thematic categories denoted by $A, B, \ldots, T$ [1]. User activity is reported as a sequence of category ids of pages displayed by user. In order to compress the data, subsequent occurrences of the same category were represented as a pair composed of category id and number of occurrences. Formally we denote such sequence as $\{c_i, m_i\}_{i=1}^{n}$, where each $c_i \in \mathcal{C}$ is a category id and m_i is a number of its occurrences; let $\mathcal{S}$ be the set of all sequences.

We are given two tables. Each of them consists of two fields: user id and user activity. First table describe users' activity during first three weeks and the

[1] In the original dataset categories were denoted by means numbers from 1 to 20. However for the sake of clarity we mapped numeric ids into letters.

J. Yu et al. (Eds.): RSKT 2010, LNAI 6401, pp. 520–527, 2010.

second describe the activity during fourth week. Our goal is to recognize user id on the basis of his activity in fourth week, having given activities of all users during the first three weeks labeled by user ids.

This is a classification task in which we have 4882 values of decision attribute (there are 4882 users) and objects are described by means of sequences.

However, the behaviour does not distinguish users, rather split them into clusters of indiscernible objects. As a consequence the classifier should indicate a set of users indiscernible with the one whose description is given as a classifier input, rather that point a single user.

Formally, describe our task as follows: we are given a set of users U, a function $g : U \to \mathcal{S}$ that describe user behaviour during the first three weeks. We are also given a multiset V of sequences generated by users during the fourth week. Our goal is to generate function $f : V \to P(U)$ that will be a classifier which indicates users indiscernible with the one who generated classified sequence.

For the classifier validation purposes we use also a function $h : U \to \mathcal{S}$ which describe user behaviour during the fourth week. h is the function approximated by classifier.

2 Data Description and Preprocessing

We analyze web traffic data from Polish web sites employing *gemiusTraffic* study. Data was acquired by Gemius company through the use of scripts, placed in code of the monitored web page. The scripts reported to the *gemiusTraffic* platform each Page View - event of displaying the monitored web page. Page Views were grouped into Visits - uninterrupted series of Page Views on a given web site executed by the same user.

This dataset was used as subject of one tasks in ECML/PKDD'2007 Discovery Challenge [3]. However the of that task was different that the one we pursue here. The problem objective was to predict user behaviour, while we are recognizing users on the basis of their behaviour.

The original dataset was composed out of two tables. First of them consisted of information about users: their country, operating system, browser, etc. This information is irrelevant for our task, so we did not use the first table.

Second table described users' activity in terms of Visits. It consisted of the following fields: user id, timestamp and sequences of categories of pages displayed during the Visit. For each user we have merged subsequent Visits into one sequence of Page Views. This operation significantly simplified the structure of data. According to our tests, the information about how Page Views are grouped into Visits is not helpful in our task.

3 Quality Measures

Before we start to construct a classifier we must define measures of its quality. Our task is different from a typical classification problem, because of the large

number of decision attribute values. That is why traditional quality measures [2] developed for classification tasks (such as accuracy) does not suit to our task.

Since the classifier indicates a set of users indiscernible with the one whose description is given as its input, the quality measures should compare the set of users indicated by the classifier with the set of users indiscernible with a given one (the latter set we denote also as a set of relevant users).

For a single set of relevant users our task is similar to the problem of information retrieval: searching for documents relevant for a given query. In this case *precision* and *recall* are adequate quality measures. They can be seen as a measures of classifier exactness and completeness.

We define *precision* as the number of relevant users indicated by a classifier divided by the total number of users indicated by a classifier, and we define *recall* as the number of relevant users indicated by a classifier divided by the total number of relevant users.

Formally, let $[u]$ denote a set of users indiscernible with $u \in U$ (equivalence class of the indiscernibility relation). For a given classifier f and user $u \in U$ we define precision and recall as follows:

$$\mathbf{Precision}(f,u) = \frac{|[u] \cap f(h(u))|}{|f(h(u))|}, \mathbf{Recall}(f,u) = \frac{|[u] \cap f(h(u))|}{|[u]|},$$

where $h(u)$ is a sequence generated by user u and $f(h(u))$ is a value calculated by classifier for this sequence.

Now, we extend these measures on the sets of users. Since precision and recall are identical for indiscernible objects it is natural to state that

$$\mathbf{Precision}(f,[u]) = \mathbf{Precision}(f,u), \mathbf{Recall}(f,[u]) = \mathbf{Recall}(f,u)$$

and precision and recall of sets that are sums of indiscernibility classes are weighted means of their values calculated separately for each indiscernibility class.

On the other hand, values of precision and recall for sets of users should be an arithmetic mean of their values calculated for each user:

$$\mathbf{Precision}(f,X) = \frac{1}{|X|}\sum_{u \in X} \mathbf{Precision}(f,u), \mathbf{Recall}(f,X) = \frac{1}{|X|}\sum_{u \in X} \mathbf{Recall}(f,u).$$

Now, we show that the above definitions are equivalent. Let $X \subset U$ be such that $X = [u_1] \cup [u_2] \cup \ldots \cup [u_n]$.

$$\mathbf{Recall}(f,X) = \frac{1}{|X|}\sum_{u \in X} \mathbf{Recall}(f,u) = \frac{\sum_{k=1}^{n} |[u_k]| \mathbf{Recall}(f,u_k)}{\sum_{k=1}^{n} |[u_k]|} =$$

$$= \frac{\sum_{k=1}^{n} |[u_k]| \mathbf{Recall}(f,[u_k])}{\sum_{k=1}^{n} |[u_k]|} = \frac{\sum_{k=1}^{n} |[u_k] \cap f(h(u_k))|}{\sum_{k=1}^{n} |[u_k]|}.$$

Analogically

$$\mathbf{Precision}(f,X) = \frac{\sum_{k=1}^{n} |[u_k]| \mathbf{Precision}(f,[u_k])}{\sum_{k=1}^{n} |[u_k]|} = \frac{\sum_{k=1}^{n} |[u_k]| \frac{|[u_k] \cap f(h(u_k))|}{|f(h(u_k))|}}{\sum_{k=1}^{n} |[u_k]|}.$$

Precision and recall may be combined into a single measure *F-score* which is their harmonic mean:

$$F(f,X) = 2 \cdot \frac{\mathbf{Precision}(f,X) \cdot \mathbf{Recall}(f,X)}{\mathbf{Precision}(f,X) + \mathbf{Recall}(f,X)}.$$

Apart from the above measures we will consider also *average cluster size* defined as

$$\mathbf{ClusterSize}(f,X) = \frac{1}{|X|} \sum_{u \in X} |f(h(u))| = \frac{\sum_{k=1}^{n} |[u_k]| \cdot |f(h(u_k))|}{\sum_{k=1}^{n} |[u_k]|},$$

where $X \subset U$ is such that $X = [u_1] \cup [u_2] \cup \ldots \cup [u_n]$.

We assume that recall is the most important quality measure: high recall guarantee that the relevant user will be among users indicated by classifier. The second important measure is the cluster size: it tells us how many users will be on average indicated by classifier.

4 Feature Selection

Users cannot be classified by classifier that simply receive sequences of Page Views as input data and tries to find dependence between user id and sequence. The reason is that such classifier would have an infinite Vapnik–Chervonenkis dimension [4], so it would not be learnable [1].

In order to construct learnable classifier, we had to select relevant features from the original data. We accomplished this goal adaptively guessing how relevant information is encoded in data: we prepared and tested several features and, on the basis of tests results, we developed feature selectors described in this and the following sections.

We assume that user preferences did not change during 4 weeks covered by data. The reason for this assumptions is the fact that 4 weeks is a very short period so there is a small probability that user will change his preferences, on the other hand users behave pretty randomly so it is very hard to distinguish preference change from random noise.

As a consequence, categories that appear in sequence generated by a given user during first three weeks should be identical to those which he generated during fourth week. However, some rare categories may appear only in one sequence as a statistical noise. In order to eliminate such noise we introduce thresholds: we eliminate from sequence all categories that were viewed less then a given number of times.

For a given threshold t function $a_t : \mathcal{S} \to P(\mathcal{C})$ selects features as follows:

$$a_t\big(\{c_i, m_i\}_{i=1}^{n}\big) = \{c : \sum_{i:c_i=c} m_i > t\} \cup \{\arg\max_{c} \sum_{i:c_i=c} m_i\}.$$

It generates a set of categories, such that user displayed more than t pages belonging to each of them. In order to avoid users described by empty feature set, the most frequent category is always added to the set.

Function a_t generates indiscernibility classes, which may be interpreted as user profiles. Here are presented the most common user profiles, generated with threshold $t = 30$:

1176 Q	80 GNQ	27 F	15 CG	11 ILQ
834 GQ	73 IQ	27 CGQ	14 NQ	11 HPQ
676 G	69 I	25 FQ	14 N	11 $GLNQ$
186 LQ	61 GP	23 CQ	14 GH	11 FG
157 GPQ	58 P	22 $GLPQ$	14 C	10 LPQ
116 GN	52 GIQ	19 $GNPQ$	13 GNP	10 FGQ
108 L	45 GHQ	17 GOQ	13 $CGPQ$	10 EHQ
99 PQ	40 H	17 EQ	12 OQ	10 EGQ
89 GLQ	36 GL	15 MQ	12 $GHPQ$	9 KQ
87 HQ	29 GI	15 GKQ	11 IPQ	9 GK

Size of profiles decreases exponentially, we have a few large clusters and many small ones: there are 245 profiles total. The number of categories in profiles is small, typical user visits pages belonging to 1, 2 or 3 categories.

We tested also alternative feature selection function $b_t : \mathcal{S} \to P(\mathcal{C})$

$$b_t\big(\{c_i, m_i\}_{i=1}^n\big) = \{c : \sum_{i:c_i=c} m_i > t \cdot \sum_{i=1}^{n} m_i\}.$$

It selects categories which are present in at least t percent of pages displayed by user.

Our feature selectors seek for information concerning the thematic categories in which user is interested. That is why, the order of visited pages is irrelevant for them and the number of pages belonging to a given category is important only in terms of being satisfying required threshold.

5 Classifier $f_{\text{simple}\subseteq}$

Now, we construct a simple classifier which will test our features selectors.

The classifier $f_{\text{simple}\subseteq} : V \to P(U)$ selects a set $a_t(s)$ of categories which are present in at least t pages in input sequence. Then it returns set of users for whom set $a_t(s)$ is a subset of a set of categories of pages visited by this user during first three weeks. It is defined as:

$$f_{\text{simple}\subseteq}(s) = \{u \in U : a_t(s) \subseteq a_0(g(u))\}.$$

Classifier has the following performance when used with features selected by a_t:

t	**Recall**	**Precision**	F	**ClusterSize**
0	0.4318	0.0119	0.0232	1211.6
1	0.6059	0.0226	0.0436	1647.3
2	0.6956	0.0312	0.0597	1910.3
3	0.7522	0.0385	0.0733	2099.7
5	0.8191	0.0504	0.0949	2356.7
7	0.8558	0.0597	0.1116	2543.1
10	0.8947	0.0721	0.1335	2747.9
15	0.9295	0.0892	0.1629	3000.6
20	0.9480	0.1044	0.1881	3177.1
25	0.9592	0.1156	0.2064	3301.4
30	0.9664	0.1268	0.2242	3405.6
35	0.9746	0.1397	0.2443	3497.0
40	0.9801	0.1512	0.2620	3567.9
50	0.9846	0.1721	0.2931	3674.5

It offers arbitrarily high recall, however it has large cluster size and small precision.

Performance of an analogical classifier that selected features according to b_t was inferior to $f_{\text{simple}\subseteq}$

Performance of classifier $f_{\text{simple}\subseteq}$ is significantly dependent from number of categories present in cluster (below, the dependence $f_{\text{simple}\subseteq}$ classifier performance from number of categories present in cluster, with threshold $t = 30$ is presented):

No cats	No users	Recall	Precision	F	ClusterSize
1	2224	0.9977	0.1797	0.3046	4276.1
2	1752	0.9657	0.1126	0.2018	3194.1
3	692	0.9031	0.0291	0.0565	1909.7
4	185	0.8648	0.0083	0.0165	1011.3
5	25	0.7600	0.0025	0.0050	420.9
6	3	0.6666	0.0059	0.0118	214.0
7	1	1.0000	0.0045	0.0091	218.0

Observe that large cluster size is a consequence of poor distinctive ability of classifier for users with small number of categories. On the other hand, for clusters specified by more than 3 categories, we observe decrease of recall. That is why, now, we will develop classifiers dedicated to clusters with a certain number of categories, then we combine them into one classifier which will work with recall 0.9.

6 Classification of Sequences Specified by One Category

In this case we know that examined user during the fourth week displayed pages belonging to only one category. The only information that may help us to indicate such user is the percent of Page Views that belong to that category during the first three weeks of user activity.

Let s be a sequence such that $a_{30}(s)$ is a singleton. Let $\mathbf{t} = (t_A, t_B, \ldots, t_T)$. The classifier $f_{\mathbf{t}} : V \to P(U)$ is defined as:

$$f_{\mathbf{t}}(s) = \{u \in U : a_{30}(s) = \{c\} \wedge c \in b_{t_c}(g(u))\}.$$

In the definition above $a_{30}(s) = \{c\}$ means that c is the only category that appears in sequence s more than 30 times and $c \in b_{t_c}(g(u))$ means that more than t_c percent of pages displayed by user u is labeled with category c.

The following table presents optimal $\mathbf{t}$ and quality of $f_{\mathbf{t}}$ classifier for single categorial clusters (Last verse describes classifier on all single categorial clusters together):

Cluster	No users	t	Recall	Precision	F	ClusterSize
G	676	0.50	0.9038	0.3790	0.5341	1612.0
H	40	0.19	0.9000	0.2000	0.3273	180.0
K	8	0.43	1.0000	0.2857	0.4444	28.0
P	58	0.03	0.9310	0.0432	0.0826	1250.0
Q	1176	0.46	0.9022	0.5247	0.6635	2022.0
joint	2224		0.9069	0.4356	0.5885	1624.1
$f_{\text{simple}\subset}$	2224		0.9977	0.1797	0.3046	4276.1

As we can see each cluster has its own specific character: clusters differs in size, threshold value and precision. Some clusters are simple to classify while others (see P-cluster for example) are pretty hard. In general $f_{\mathbf{t}}$ reduced average cluster size from 4276.1 to 1624.1 maintaining recall at the level 0.9.

7 Classification of Sequences Specified by Two Categories

Consider the following sequences:

$$CQCGQGQGQGQGQGQGQGQC,$$
$$FGFGFGFGFGFGFGFGQFQF,$$
$$FGFIFIFGFGFGFIF.$$

They consists of long periods of alternating occurrences of two categories. We observed that mosts of sequences have such structure.

We recognize such periods when they have at least four occurrences and we replace recognized periods with new categories. Each occurrence of such new category is assigned with the number of Page Views equal to the sum of Page Views that appear in recognized period.

For example the above sequences are processed into:

$$CQC\,\underline{GQ}\,C,\ \ \underline{FG}\,\underline{FQ},\ \ FG\,\underline{FI}\,\underline{FG}\,IF.$$

While classifying sequence specified by two categories (we will denote them as c_1 and c_2) we considered following features:

- the quantity of occurrence of c_1 category in sequence,
- the quantity of occurrence of c_2 category in sequence,
- the quantity of occurrence of c_1c_2 category created from alternating occurrences of c_1 and c_2.

We analyzed performance of 4 classifiers:

$$f_{\mathbf{t},\mathbf{t}'}(s) = \{u \in U : a_{30}(s) = \{c_1, c_2\} \wedge c_1 \in b_{t_{c_1c_2}}(g(u)) \wedge c_2 \in b_{t'_{c_1c_2}}(g(u))\},$$
$$f_{\mathbf{t}''}(s) = \{u \in U : a_{30}(s) = \{c_1, c_2\} \wedge c_1c_2 \in d_{t''_{c_1c_2}}(g(u))\},$$
$$f_{\mathbf{t},\mathbf{t}'\wedge\mathbf{t}''}(s) = f_{\mathbf{t},\mathbf{t}'}(s) \cap f_{\mathbf{t}''}(s),$$
$$f_{\mathbf{t},\mathbf{t}'\wedge\mathbf{t}''}(s) = f_{\mathbf{t},\mathbf{t}'}(s) \cup f_{\mathbf{t}''}(s).$$

Performance of $f_{\mathbf{t},\mathbf{t}'}$ and $f_{\mathbf{t}''}$ that both quantity of single categories occurrences and quantity of alternating occurrences of categories carry important information. $f_{\mathbf{t},\mathbf{t}'\wedge\mathbf{t}''}$ and $f_{\mathbf{t},\mathbf{t}'\wedge\mathbf{t}''}$ tested possible combinations of this information. Test showed that $f_{\mathbf{t},\mathbf{t}'\vee\mathbf{t}''}$ classifier gives best results.

A possible reason of such behaviour is that quantity of single categories occurrences is relevant feature for short sequences and quantity of alternating category occurrences is relevant for long sequences. So, these features operate on disjoint sets of sequences and should be connected by disjunction.

The following table presents optimal parameters and performance of $f_{\mathbf{t},\mathbf{t}'\vee\mathbf{t}''}$ classifier for clusters that have more then 10 users:

Cluster	No user	t	t'	t"	Recall	Precision	F	ClusterSize
CG	15	0.01	0.01	0.88	0.9333	0.0648	0.1212	216.0
CQ	23	0.22	0.02	0.33	0.8261	0.0662	0.1226	287.0
GN	116	0.95	0.48	0.05	0.9052	0.4217	0.5753	249.0
GP	61	0.01	0.66	0.00	0.9016	0.0469	0.0891	1173.0
GQ	834	0.09	0.11	0.37	0.9005	0.2950	0.4444	2546.0
HQ	87	0.45	0.01	0.15	0.9080	0.0607	0.1138	1302.0
MQ	15	0.37	0.23	0.11	0.8000	0.4286	0.5581	28.0
PQ	99	0.79	0.00	0.04	0.9091	0.0317	0.0612	2841.0
joint	1667				0.9010	0.2431	0.3829	1714.0
$f_{\text{simple}\subseteq}$	1752				0.9657	0.1126	0.2018	3194.1

Last verse describes performance of classifier on all two categorial clusters together. Cluster have their own specific character similarly as single categorial clusters. For some of them, it was impossible to find recall greater then 0.9. Classifier reduced average cluster size from 3194.1 to 1714.0 maintaining recall at the level 0.9.

In order to avoid overfitting we used $f_{\text{simple}\subseteq}$ for all clusters that have less then 11 users.

8 Results

We defined our ultimate classifier f_{combined} as follows: for a given sequence, we select relevant categories using feature selector a_{30}, then we check the number of categories in obtained set:

- if there is only one category, we classify sequence using $f_{\mathbf{t}}$ classifier with parameters specified in section 6;
- if there are two categories and this pair of categories is present in table in section 7, we classify sequence using $f_{\mathbf{t},\mathbf{t}'\vee\mathbf{t}''}$ classifier with parameters specified in that table;
- otherwise we classify sequence by means of $f_{\text{simple}\subseteq}$ classifier.

Classifier f_{combined} has the following performance for our set of users U:
$\mathbf{Recall}(f_{\text{combined}}, U) = 0.9015, \mathbf{Precision}(f_{\text{combined}}, U) = 0.2860,$
$F(f_{\text{combined}}, U) = 0.4342, \mathbf{ClusterSize}(f_{\text{combined}}, U) = 1651.9.$

Acknowledgment. The research has been partially supported by grants N N516 368334 and N N516 077837 from Ministry of Science and Higher Education of the Republic of Poland.

References

1. Angluin, D.: Inductive inference of formal languages from positive data. Information and Control 45, 117–135 (1980)
2. Guillet, F., Hamilton, H.J. (eds.): Quality Measures in Data Mining. Studies in Computational Intelligence, vol. 43. Springer, Heidelberg (2007)
3. Jaworska, J., Nguyen, H.S.: Users Behaviour Prediction Challenge. In: Nguyen, H.S. (ed.) Proc. ECML/PKDD 2007 Discovery Challenge (2007)
4. Vapnik, V.N.: Statistical Learning Theory. Wiley, New York (1998)