Temporal evolution of the UK Web*

Ilaria Bordino
Sapienza Università di Roma
Rome, Italy
bordino@dis.uniroma1.it

Paolo Boldi
Università degli Studi di Milano
Milan, Italy
boldi@dsi.unimi.it

Debora Donato
Yahoo! Research
Barcelona, Spain
debora@yahoo-inc.com

Massimo Santini
Università degli Studi di Milano
Milan, Italy
santini@dsi.unimi.it

Sebastiano Vigna
Università degli Studi di Milano
Milan, Italy
vigna@dsi.unimi.it

Abstract

Recently, a new temporal dataset has been made public: it is made of a series of twelve 100M pages snapshots of the `.uk` *domain [2]. The Web graphs of the twelve snapshots have been merged into a single* time-aware *graph that provide constant-time access to temporal information. In this paper we present the first statistical analysis performed on this graph, with the goal of checking whether the information contained in the graph is reliable (i.e., whether it depends essentially on appearance and disappearance of pages and links, or on the crawler behaviour). We perform a number of tests that show that the graph is actually reliable, and provide the first public data on the evolution of the Web that use a large scale and a significant diversity in the sites considered.*

1 Introduction

Understanding and analyzing the structure and the evolution of the Web is an important and delicate challenge that requires theoretical efforts of modelization to be always corroborated by empirical findings. The latter, in turn, are costly to obtain, because they require bandwidth, computation time and human intervention, besides to robust software to gather the data and to provide easy access to the collected information. Indeed, apart for commercial search engines, there have been only very few attempts to perform such a task in an academic setting, and to make the data publicly available. The only previous project with this aim is the Internet Archive[1], a non-profit organization that is trying to provide a sort of time collection of the Web; their dataset cannot be easily accessed for batch analysis, and although socially and historically important, it is of scarce interest for those who aim at studying structural properties.

The problem becomes even more elusive if time evolution is taken into account, because one would like to have not only different snapshots of the same portion of the Web to be available, but also some guarantee on their mutual consistency (for example, to be sure that the same crawling policies have been followed) is in that case of imperative importance.

In this paper, we analyze the data obtained by an experiment performed during one year that consisted in gathering twelve 100M pages snapshots of the `.uk` Web, at a monthly rate. In particular, we focus our attention on the problem of assessing the data so obtained and understanding its anomalies found: this is of uttermost importance if one wants to understand correctly how the dataset evolved in time.

The rest of the document is organized as follows. Section 2 presents related work. Section 3 describes how the data were collected and the time-aware graph was built. Section 4 presents the set of experiments we conducted to assess the reliability of the graph. In section 5 we study the temporal evolution of the data collection. Section 6 is a short conclusion.

2 Related work

Several previous works focused on Web evolution. Cho and Garcia Molina [5] performed a daily collection of around $720,000$ pages from 270 popular sites during a period of 4 months. They counted how many days each URL

*This work was partially supported by the EU within the 6th Framework Programme under contract 001907 “Dynamically Evolving, Large Scale Information Systems” (DELIS)

[1] http://www.archive.org/

was accessible and proposed estimators for the frequency of change of Web pages.

Brewington and Cybenko [4] collected pages observed over an average interval of 37 days and used the recording of the *last-modified* timestamp and the downloading time of Web pages to study their rate of change.

Fetterly *et al.* [6] crawled on a weekly basis a set of 150 million URLs obtained from the Yahoo! home page, spanning a time interval of 11 weeks in 2002. This work aimed at studying the frequency and the degree of change of Web pages. The authors observed that a significant amount of changes on the Web consists of small modifications, like html tags.

Ntoulas *et al.* [9] collected for one year exhaustive weekly downloads of Web pages belonging to 154 popular sites gathered from the Google Directory. They observed high birth and death rates for Web pages, and an even higher turnover rate for the hyperlinks. At the same time, they noticed that newly created pages tend to borrow their content heavily from existing pages, and most of the pages that persist over time exhibit only minor changes in their content.

Koheler [8] considered a collection of 361 URLs selected at random from a Web crawl during a period of 4 years. The author performed accessibility tests, observing a periodic resurrection of Web pages and sites. He claimed that once a collection has reached a considerable age, it tends to stabilize.

Gomes and Silva [7] examined a set of 51 million pages, using a number of snapshots gathered from a national community web, spanning a period of 3 years. The authors measured the persistence of both URLs and Web content. They found that most URLs have a short life, while a minor fraction of pages persist for long periods of time. Gomes and Silva observed that persistent URLs are static, short and tend to be linked from other sites.

Toyoda and Kitsuregawa [10] analyzed data from the Japanese Web Archive and proposed a novelty measure for estimating the certainty that a newly created page newly appeared within a series of unstable snapshots. This measure can be used to extract novel pages for further analysis with reasonable precision.

3 Collecting the dataset

This section describes how we built our data collection; we just recall the main points described in [2]. The dataset consists of a large time-aware web graph containing the graphs of twelve monthly snapshots of the `.uk` domain. All the graphs are freely downloadable[2]. The data were collected during one year, from June, 2006 to May, 2007, with monthly crawls by the LAW (Laboratory for Web Algorithmics) of the Università degli Studi di Milano.

The graph was built using WebGraph [3], a framework for web graph compression that currently provides the best compression available in terms of bits per link. For the merged graph, WebGraph was augmented with the possibility of storing highly compressed labels on the arcs.

3.1 Crawling parameters

The snapshots have been taken at the start of each month, during a period of $7 - 10$ days. Some basic information about the snapshots is shown in Table 1. A careful definition of the crawling parameters is crucial, as in any limited-size crawl. The stopping criterion is clearly that of reaching about 100M pages, without counting duplicates. The main features are listed below.

Crawl policy. We use UbiCrawler's [1] built-in per-host breadth-first visit. A number of threads scan in parallel distinct hosts, and newly discovered URLs are added to a queue. When a thread completes its visit, it extracts from the queue the first URL whose host has an IP address that is not currently being visited, and starts visiting that host in a breadth-first fashion.

Seed. The seed is a large (190 000 elements) set of URLs obtained from the Open Directory Project [3]. The reason for such a large seed is that of making the crawl more stable and repeatable, and reducing the amount of spam (as links in the Open Directory Project are judged by humans).

Maximum number of pages per host. We limited each host to a maximum of 50 000 pages. This guarantees that we shall crawl at least 2 000 hosts, and limits the impact of web traps and database-driven sites.

Maximum inter-host depth. We do not delve more than 16 levels in a host. The main reason for a limit in depth is avoiding traps and also badly configured 404 pages, which sometimes generate an infinite number of links by prefix buildup.

URL normalisation. URLs are normalised following the strategy explained in the `BURL`[4] Java class. We apply all safe normalisations, escape all illegal characters, and treat in a special way square brackets as they are ubiquitously (although erroneously) used in an unescaped form.

Duplicate detection. Many pages are duplicates, and to detect their presence we maintain a set of 64-bit fingerprints obtained after stripping attributes (of HTML elements) and other non-relevant parts of the page. When

[2] `http://law.dsi.unimi.it/`

[3] www.dmoz.org

[4] The class is available in bundle with the LAW software, downloadable at `http://law.dsi.unimi.it/`.

	Pages	GZip'd Size (GB)	Nodes	Arcs	Graph Size(GB)	bit/arc
Jun	112 386 763	402	80 644 902	2 481 281 617	0.89	3.07
Jul	136 956 559	477	96 395 298	3 030 665 444	1.16	3.30
Aug	141 395 895	507	100 751 978	3 250 153 746	1.23	3.25
Sep	148 965 298	546	106 288 541	3 871 625 613	1.32	2.93
Oct	129 558 491	478	93 463 772	3 130 910 405	1.03	2.83
Nov	150 146 132	546	106 783 458	3 479 400 938	1.16	2.86
Dec	144 489 446	525	103 098 631	3 768 836 665	1.34	2.77
Jan	151 578 113	553	108 563 230	3 929 837 236	1.38	2.72
Feb	153 966 540	564	110 123 614	3 944 932 566	1.39	2.74
Mar	151 427 461	545	107 565 084	3 642 701 825	1.34	2.84
Apr	150 606 689	559	106 867 191	3 790 305 474	1.36	2.79
May	150 054 551	556	105 896 555	3 738 733 648	1.30	2.69

Table 1: Per-snapshot full-text and web-graph stats.

a duplicate is detected we just store a pointer to the original page. About 25% of the overall pages happen to be duplicates (so, to collect 100M distinct pages we had to download some 130M pages at each crawl).

3.2 Aligning the Snapshots

The first important step in getting a temporally labelled collection is *alignment*: identifying URLs in different snapshots that correspond to the same Web page. Alignment is a non-trivial issue because if a URL is not static it might contain session-generated data (e.g., a session ID) that makes *de facto* identical URLs appear as syntactically distinct. For the present collection, the radical choice was made of considering only static URLs (i.e., URLs that do not contain a question mark[5]). Table 2 shows that, as a first attempt, our choice is not unreasonable. However, one of the open problems is to develop some sensible alignment technique for dynamic URLs.

3.3 Assigning temporal labels to the Graph

Once URLs are aligned, we obtain a *global* graph G that includes all static pages (and related links) appearing in each snapshot. The graph G is labelled so that, for each node and each link, we can detect whether it was present or not in any given snapshot. Given that there are twelve crawls, one per month, the graph must provide twelve bits of information per node and per arc. The labelling facilities of WebGraph, combined with a compression technique based on Huffman codes, make it possible to store a label in just 2.16 bits per label (for more details, see [2]).

4 Data Assessment

In this section we present a set of experiments performed to assess the reliability of our dataset. This assessment phase is instrumental in the subsequent usage of the crawled data, and aims at establishing how confident we can be that a snapshot itself is a faithful representation of the `.uk` domain. We first provide details about the definitions and the notation used in the experiments. Subsequently, we present the experiments themselves and the relative results.

4.1 General definitions and notation

Let T be the number of snapshots collected, numbered from 1 to T; the t-th snapshot consists of:

- a set S_t of *seen URLs* that includes all URLs ever found by the crawler;
- a set C_t of *crawled URLs*[6]: $C_t \subseteq S_t$ this is the set of URLs that have a WARC record in the WARC store saved by the crawler; such set is divided into:
  - a set V_t of *existing URLs*, which form the nodes of the t-th crawl graph;
  - a set F_t of *failure URLs*: this is the set of URLs that resulted in a 4xx response;
  - a set D_t of *duplicate URLs*: the crawler saved them as duplicate of some other existing URL (an element of V_t), called its *archetype*; in other words, there is a map $a_t : D_t \to V_t$ that maps each duplicate URL to its archetype. For convenience, let us extend it on $D_t \cup V_t$ by the identity.

[5] This choice, unfortunately, cannot prevent *opaque* session-dependent URLs from generating noise in the collection.

[6] We highlight here the distinction between a URL and the content of the page it refers to: in this document we shall try to keep this distinction in mind and coherently use URL and page in the different contexts.

	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May
Jun	31.3	19.0	18.3	17.2	15.3	15.0	13.8	13.7	13.2	12.3	11.9	11.1
Jul		35.2	23.3	21.7	18.5	18.3	16.2	16.4	16.0	15.2	14.5	13.6
Aug			37.3	24.3	19.4	19.4	17.1	17.3	16.7	15.5	15.0	13.8
Sep				39.9	21.2	21.2	18.7	19.0	18.1	16.9	16.3	14.7
Oct					33.8	22.2	19.0	19.1	18.3	16.6	16.4	15.0
Nov						37.3	22.3	21.9	21.3	18.8	18.3	16.6
Dec							36.6	23.5	21.0	19.0	17.9	16.6
Jan								39.0	23.7	20.8	19.9	18.4
Feb									37.7	23.1	22.2	20.0
Mar										38.1	22.4	20.2
Apr											36.9	24.2
May												36.9

Table 2: Static URLs overlap, in millions.

- for every crawled URL $u \in C_t$, we have a stored page $P_t(u)$, that is, the content and HTTP headers received and saved in the WARC file (note that the crawler does not save the content for duplicate URLs, so for the sake of definiteness we let $P_t(u) = P_t(a(u))$ whenever $u \in D_t$).

The t-th crawl graph is the directed graph $G_t = (V_t, E_t)$ where $(x, y) \in E_t$ if $P_t(a_t(x))$ contains an anchor y' with $y = a_t(y')$.

We also let $C = \cup_t C_t$, $F = \cup_t F_t$, $V = \cup_t V_t$, $D = \cup_t D_t$, $E = \cup_t E_t$ and $G = (V, E)$.
For every $u \in V$, consider the map $f_u : \{1, \dots, T\} \rightarrow \{0, 1\}$ such that $f_u(t) = 1$ iff $u \in V_t$: this is what we will refer to as *page appearance trace (PAT)*[7].

The *persistence* of a URL between times i and j is the fraction of 1s in its PAT between positions i and j.

4.2 Preprocessing: Classifying URLs

This experiment aims at classifying the URLs in $u \in V$, according to the following parameters:

- *crawl depth*(D_C): the length of the shortest directed path from a(ny) URL in the seed to u; more precisely, u has depth 0 if it is in the seed[8], and it has depth $k+1$ if there is some u' of depth k such that $(u', u) \in E$;
- *syntactic depth*(D_S): the number of slashes "/" in the path part of the URL.

The left side of Table 3 shows the head of the crawl depth distribution, reporting, for the most common values of the parameter, the percentage of nodes with that specific depth value. It is worth to note that the fourth most common value is equal to Infinity. Nodes that have an infinite crawl depth cannot be reached from the seed. There are almost 16 million such nodes, that is, a fraction equal to 11% of the total nodes. This means that, by considering only the static URLs, we are significantly disconnecting the graph. Figure 1 shows the crawl depth distribution (only finite values are considered). We notice that the 70% of the nodes have a crawl depth not greater than 10, and the 60% of the nodes have a crawl depth not greater than 7. Hence, a major part of the nodes in the dataset are reachable from a URL in the seed with a small number of hops.

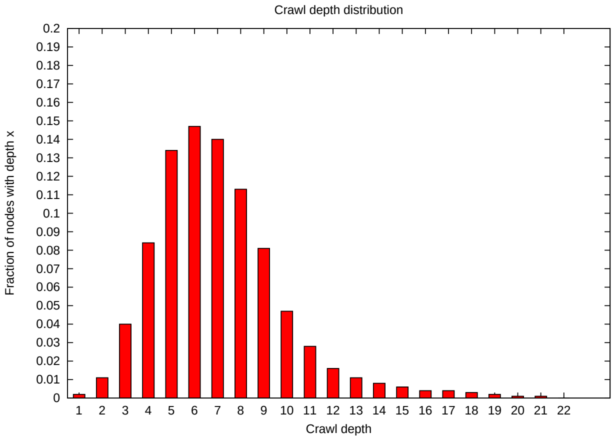

Figure 1: Crawl depth distribution

The right side of Table 3 shows the head of the syntactic depth distribution, which is plotted in Figure 2. We observe that 95% of the nodes have a syntactic depth not greater than 9, and 85% of the nodes correspond to URLs with no more than 5 syntactic levels. This is as expected, because relevant content is often located at the top of a site's hierarchy, rather than in a deeper location within the site.

The computation of the Pearson correlation coefficient between the two depth distributions returned a value equal to -0.32, which basically means that such distributions are

[7] To simplify notation, we shall write a PAT as the ordered sequence of its values.

[8] The seed was the same for all snapshots.

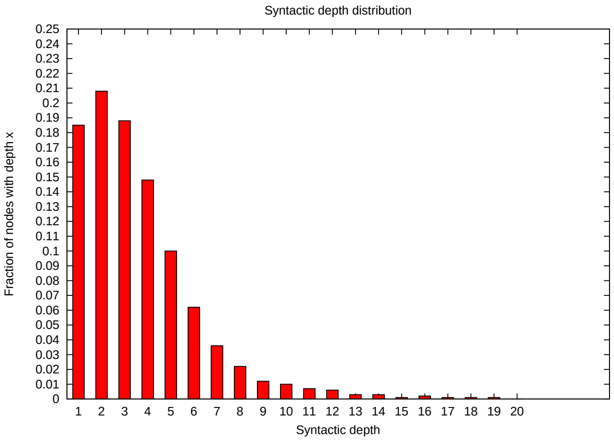

Figure 2: Syntactic depth distribution

not significantly correlated. We might expect some correlation very likely to be found, if we thought that, the deeper a page within the tree directory of a site, the longer the path to reach it starting from a page in the seed set. We believe that the observed evidence can partially be explained with the way our crawler works: it enters a site as soon as it finds some links to it, not necessarily from its homepage.

4.3 Measuring page persistence

The persistence of a page is the fraction of 1s in its PAT. We compute such a measure for the pages in the intersection between the first and the last crawl. Figure 3 shows the distribution of persistence values.

If the crawling were perfect, a page existing both in the first month and in the last one should be present in every snapshot, so we would expect its persistence to be equal to 1. This does not happen in every case: sometimes the crawler failed to crawl existing URLs. One explanation for this fact might be a temporary unavailability of a site. In some other cases, the crawler did not reach the URL because the per-host limit imposed had been exceeded. Hence, there exist nodes in the intersection between the first and the last month that are not appearing in some intermediate snapshots.

However, we observe that around 90% of the considered nodes have a persistence in the range (0.8,1). These nodes have at most two missing zeros in their PATs. We feel confident in the fact that such pages existed also in the months in which the crawler failed to capture them.

4.4 Experiments related to deletion of pages

This section provides details about the experiments we conducted to evaluate how faithfully the crawled data is able to capture the event of a page deletion.

D_C	$\%\lvert V\rvert$	D_S	$\%\lvert V\rvert$
6	15	1	19
7	14	2	21
5	13	3	19
∞	11	4	15
8	11	5	10
4	8	6	6
9	8	7	4
10	5	8	2
3	4	9	1
11	3	10	1

Table 3: Crawl depth and syntactic depth distribution.

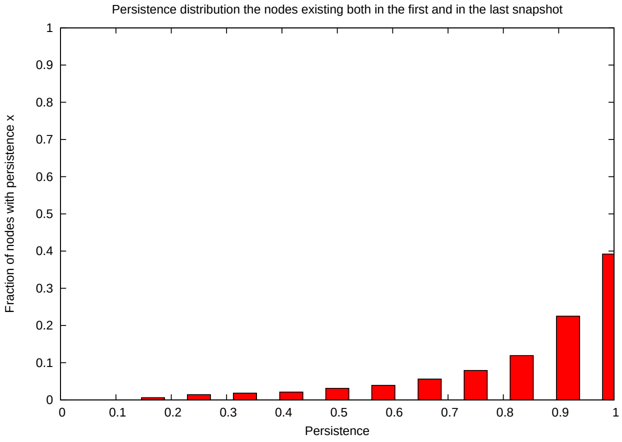

Figure 3: Persistence distribution for the nodes in the intersection between the first and the last snapshot.

4.4.1 No-Resurrection Assumption Experiment

We say that a node u *breaks the no-resurrection assumption at time* t if and only if there are $t_1 < t < t_3$ such that $f_u(t_1) = f_u(t_3) = 1$ and $f_u(t) = 0$. We believe that if a URL breaks the no-resurrection assumption at time t then it is the crawler that failed to crawl the URL for the two reasons that were mentioned in the previous subsection. A PAT breaking the no-resurrection assumption is referred to as *anomalous* PAT.

We use presence/absence information provided by node labels to compute the number of URLs that break the no-resurrection assumption. We start our analysis of the results by dividing the appearance traces of the nodes into four groups:

- *Monotone non descending (A):* the nodes that appeared at some time and then never disappeared. The PAT of these nodes has the form $\mathtt{0}^*\mathtt{1}^+$.
- *Monotone descending (D):* the nodes that appeared in the first snapshot and at some point disappeared for-

ever. These nodes are characterized by a PAT like $1^{+}0^{+}$.

- *Plateaux (P):* the nodes that appeared at some time and then disappeared forever. Such nodes exhibit a PAT with the form $0^{+}1^{*}0^{+}$.
- *Bad nodes (B):* the nodes that break the no-resurrection assumption. These pages disappeared at some point and then reappeared.

Pattern type	% $\|V\|$
A	12.35
D	11.40
P	46.44
B	29.82
Anomalous PAT	% $\|V\|$
000000000101	0.35
110111111111	0.30
101000000000	0.29
111110111111	0.28
110100000000	0.27
000000001010	0.27
010100000000	0.25

Non-desc. PAT	% $\|V\|$	% $\|A\|$
000000000001	4.74	38.41
111111111111	3.27	26.48
000000000011	0.98	7.91
000000000111	0.72	5.82
000000001111	0.65	5.30
011111111111	0.43	3.44
000000011111	0.38	3.10
000000111111	0.33	2.70
000111111111	0.23	1.88
000011111111	0.22	1.81
001111111111	0.21	1.69
000001111111	0.18	1.45

Table 4: (a) Appearance traces for all the nodes in the graph and common anomalous PATs.(b) Monotone non descending appearance traces.

The upper section of Table 4(a) shows the percentage number of occurrences of each type of pattern. All the possible $2^{12}-1$ patterns actually appear in the data. We observe that 12.35% of the nodes exhibit monotone non descending appearance traces, while 11.40% of the Web pages in the union graph have a PAT characterized by a monotone descending behavior. Monotone PATs are reported in Tables 4(b) and 5(a). The 44.46% of the nodes are classified as *plateaux*: they appear at a given time, they exist in a number of consecutive crawls, then they disappear forever. The number of bad nodes is less than 30%. The lower section of Table 4(a) reports the percentage number of occurrences of the most common anomalous PATs. It is worth to note that these patterns contain only one *misplaced* 0. The pages characterized by such appearance traces do exist in a number of consecutive crawls, except for one month.

Desc. PAT	% $\|V\|$	% $\|D\|$
100000000000	5.25	46.06
110000000000	2.01	17.63
111000000000	0.87	7.64
111111111110	0.67	5.88
111100000000	0.56	4.94
111111110000	0.46	4.00
111110000000	0.42	3.67
111111000000	0.35	3.05
111111100000	0.34	3.02
111111111000	0.29	2.50
111111111100	0.18	1.61

Time	% $\|B\|$	% $\|V\|$
1	0.00	0.00
2	14.16	4.22
3	25.68	7.66
4	27.80	8.29
5	31.49	9.39
6	36.47	10.87
7	31.93	9.52
8	35.74	10.66
9	24.95	7.44
10	22.22	6.63
11	14.92	4.45
12	0.00	0.00

Table 5: (a) Monotone descending appearance traces.(b) Change of the fraction of nodes breaking the no-resurrection assumption as t changes.

Depth	1	2	3	6	7	9	10
t = 1	0.04	0.14	0.77	1.74	1.46	0.60	0.37
t = 2	0.05	0.25	0.87	1.86	1.54	0.83	0.48
t = 3	0.04	0.23	0.79	2.01	1.73	0.85	0.56
t = 4	0.05	0.25	0.83	2.49	2.04	0.93	0.62
t = 5	0.04	0.21	0.76	2.04	1.64	0.77	0.48
t = 6	0.04	0.24	0.85	2.38	1.97	0.89	0.60
t = 7	0.03	0.21	0.83	1.13	0.88	0.85	0.53
t = 8	0.04	0.24	0.86	1.85	1.64	1.02	0.54
t = 9	0.04	0.25	0.92	2.38	1.97	0.93	0.57
t = 10	0.04	0.25	0.83	2.16	1.74	0.88	0.55
t = 11	0.04	0.22	0.81	2.13	1.76	0.84	0.53
t = 12	0.04	0.24	0.87	2.07	1.72	0.83	0.43

Table 6: Change of the fraction of bad nodes for different values of the crawl depth.

We also investigate how the fraction of URLs that break the no-resurrection assumption at time t changes as t changes. The results of this experiment are presented in Table 5(b), which reports, for each time t, the number of nodes that break the no-resurrection assumption at time t, expressed as percentage with respect to the number of bad nodes and to the total number of nodes. We remind here that, by definition, a node u breaks the no-resurrection assumption at time t if and only if there exist $t_1 < t < t_3$ such that u appears in months t_1 and t_3, while it is not present in the t-th snashot. There cannot be nodes breaking the assumption neither in month 1 nor in month 12. The number of bad nodes is higher in the intermediate snapshots, while it is significantly lower in months 2 and 11. This finding is not in contrast with the results provided in Table 4(a), where we observe that the first and the third most common bad traces are the ones characterized by a single bad zero in the eleventh and in the second position. The reason is that the results presented in Table 5(b) are obtained by aggregating, for each time t, the occurrences of all the PATs that contain a 0 in position t while having at least one 1 occurring in a position $t_1 < t$ and at least one 1 in a position $t_3 > t$.

We complete the present experiment by studying how the fraction of bad nodes changes depending on the URL classification parameters. Table 6 provides results for some frequent values of the crawl depth. For a given depth value k, and for every t, the percentage number of nodes breaking the no-resurrection assumption is reported. If we analyze the results obtained for a specific value of the crawl depth, we observe that the fraction of bad nodes keeps basically stable over the twelve snapshots. On the converse, if we take into account a single snapshot, we notice that, in many cases, the fraction of bad nodes is higher for higher crawl depth values. We think this is completely reasonable, because a bad behavior is more likely to be expected when we go far from the seed set, which contains well mantained pages obtained from ODP.

4.4.2 Gone-Is-Gone Assumption Experiment

Every non-anomalous PAT f_u has the form $0^k1^h0^\ell$: if $\ell > 0$ then we say that u *disappeared at time* $k + h$. Although there is in general no way to see (from the WARC files) if the URL actually disappeared, we may have a clue in some cases as follows:

- suppose that u disappeared at time t;
- let $A = \{v \in V_t \cap V_{t-1} \mid (v, u) \in E_{t-1}\}$ be the set of nodes that existed both at time t and time $t - 1$, and that were in-neighbors of u at time $t - 1$ (we assume that both v and u do not break the no-resurrection assumption);
- if $P_{t-1}(v) = P_t(v)$ (that is: if the content of page v did not change from time $t - 1$ to time t), then page $P_t(v)$ should still contain an anchor to u, which means that we should have requested u at time t and should have obtained a 4xx as answer. That is, $u \in F_t$.

A URL that does not satisfy the property above is said to *break the gone-is-gone assumption*.
We compute the number of URLs breaking the gone-is-gone assumption as follows. Let $d(u)$ be the disappearance time of u: in particular, let this be $T + 1$ if u did not disappear. We first output a list of triples (v, u, t) such that $(d(v) > d(u) = t$ and $(v, u) \in E_{t-1})$. For each such triple, we check whether it satisfies the constraint $(u \in F_t$ or $P_t(v) \neq P_{t-1}(v))$. This can be done by accessing the WARC files to see if u was actually requested but produced a 4xx, or if the content of v was changed. If the specified constraint is not satisfied, then u breaks the gone-is-gone assumption.

Table 7 presents the outcome of the experiment. The fraction of triples for which u produced a $4xx$ is very small, and the number of triples breaking the gone-is-gone assumption is also small. In most of cases, the results are what we expected: whenever the edge from node v to node u does not exist anymore, we find that the content of the page has changed. We believe this is a sign of the good quality of the adopted crawling scheme. Clearly, due to the choice of a time granularity of one month, we cannot have full confidence in the fact that our time-aware graph captures all the changes that actually occurred in the real data: some intermediate changes might be not represented. Moreover, we cannot really know how fast the content of Web pages changed from one snapshot to the subsequent one: the chosen granularity makes us unable to tell which week (day) of the considered month occurred in. The choice of a finer granularity would improve the ability of providing a faithful representation of the data.

Time	$\%u \in F_t$	$\%u \notin F_t$ $P_t(v) \neq P_{t-1}(v)$	% others
1	0.55	88.37	11.07
2	0.39	82.87	16.74
3	0.40	81.91	17.69
4	0.16	89.51	10.33
5	0.15	89.15	10.70
6	0.69	92.59	6.71
7	0.04	88.24	11.73
8	0.10	82.99	16.91
9	0.15	84.41	15.44
10	0.26	77.57	22.17
11	0.89	65.02	34.09

Table 7: Results for the gone-is-gone experiment.

4.5 Adjusting the graph

The experiments presented in the previous subsection show that the fraction of URLs that break the no-resurrection assumption is small, in particular at small depths. Also, the most common anomalous PATs contain a single anomalous extra 0.

Given that these results make us confident in the good quality of the crawling, we decide to "adjust" the graph by eliminating all anomalous PATs through the adoption of a *fill-in-the-gap* approach, that is, we make all anomalous PATs of the form $0^k1\alpha10^h$ into $0^k1^{T-(k+h)}0^h$: in other words, we assume that the URL existed also if it was not crawled. We perform the adjustment using the labelling facility implemented in WebGraph to generate new labels[9] for our time-aware Web graph. We then investigate how much

Pattern type	$\% \lvert V \rvert$
A	16.18
D	14.19
P	50.54
B	19.09
Anomalous PAT	$\% \lvert V \rvert$
000000001001	0.22
000100100000	0.17
000001001000	0.16
100100000000	0.15
001001000000	0.14
000010001000	0.13
010010000000	0.12

Table 8: Appearance traces for the all the nodes in the adjusted graph and common anomalous PATs.

[9] The new labels will be freely available.

the applied adjustment changes the outcomes of the experiments presented in the previous subsection. The upper section of Table 8 shows the number of occurrences for the various pattern types in the patched graph. We observe that the fraction of bad nodes is now equal to 19.09%: thanks to the applied adjustment, around 10% of the nodes, that were originally charaterized by an anomalous PAT, are now *good* nodes belonging to a different category (monotone non descending, monotone descending or plateaux). The lower section of Table 8 shows the most common appearance traces for the nodes that still break the no-resurrection assumption in the adjusted graph. Once the patterns with a single bad zero have been patched, the most common patterns become the ones with two bad zeros.

5 Page and link dynamics

Inspired by previous work [9], we try to quantify the turnover rate of Web pages and links. This is an aspect of potential interest to search engines designers, which have to deal with the highly dynamic nature of the Web in order to provide users with the most up-to-date results. We believe that a potential limit of the mentioned study is to be found in the fact that it was conducted on 154 Web sites collected by picking up the top-ranked pages from a subset of the topical categories of the Google Directory [10]. Such a dataset cannot be considered a true sample of the Web itself, which is mostly composed by pages that are often not well maintained. Our dataset is characterized by a huge size (133 million pages and 5 billion links) and it has been collected by performing an extensive crawling of a real Web domain. Hence, it is a much more realistic sample of the Web.

In this section we present the preliminary results we obtained by computing the same statistics proposed by Ntoulas *et al.* [9] about birth and death of web pages and links. Future work will comprise a complete study of the dynamics of the content of web pages.

5.1 Birth rate of Web pages

First of all, we examine how many new pages are created every month. Figure 4 shows, for each snapshot, the fraction of its pages that are not present in the previous crawls. We notice that the monthly birth rate of web pages varies within the range $[17\%, 35\%]$. The average rate is about 30%. Ntoulas *et al.* observed an average weekly birth rate equal to 8%. If we consider that a 8% of new pages per week corresponds to a 32% of new nodes after one month, we can observe that our results are aligned with the findings presented by [9].

Ntoulas et al. used the outcome of this experiment to make conjectures about the size of the entire Web, motivating their claims with the fact that they collected exhaustive weekly downloads of 154 popular sites, by taking with a breadth first search strategy all the reachable pages in each site. By contrast, we cannot make inferences on how fast the whole Web is growing because of our choice of collecting the snapshots using the stopping criterion of reaching 100M pages.

5.2 Birth, death and replacement of Web pages.

We next quantify how many new pages are created and how many disappear over time. We also measure which fraction of pages is replaced with new pages after a given amount of time. Figure 5 shows the number of pages that are captured in the first snapshot and still remain in the n-th one, and how many pages from the n-th crawl do not exist in the first one. The bars are normalized so that the number of pages in the first month is equal to 1. The red bars represent the pages from the first month that are still available in each given snapshot. The green bars represent the pages that do exist in a given snapshot, but are not present in the first crawl.

We notice that after one month around 65% of pages existing in the first month are still available. Such a percentage becomes equal to 45% after six months, while 30% of first-month pages also appear in the last crawl. These results suggest that existing pages are replaced by new pages at a rapid rate. Figure 6 shows a normalized version of the previous plot: the numbers related to each month are now normalized to one. We notice that after six months about 45% of the pages are pages that also appeared in the first snapshot, while 55% are *new*, in the sense that they did not exist in the first month. In the last snapshot, i.e., after one year, 35% of the pages that made their first appearance in the first crawl are still available, while 65% were not captured in the first month. These results are aligned to the ones obtained by Ntoulas *et al.* in [9], that have been computed on a dataset composed by popular and well maintained sites. We believe that this similarity in the results is another validation for our data collection and we feel confident in the fact that the time-aware graph has been built with a good accuracy and it can be used to study the temporal evolution of the Web.

5.3 Link structure evolution

The last experiment we present aims at studying how much the link structure changes over time. We analyze the birth and death rates of the hyperlinks. Figure 7 shows the fraction of edges that are newly created in each month. We

[10] directory.google.com

quantify how many new links appear in each month and how many links from the first snapshot are also present in the subsequent snapshots, comparing them against the links that are newly created. Figure 8 shows the number of links that make their first appearance in each month, normalized with respect to the number of links existing in the first month. The red portion of each bar shows the number of links from the first month that are present in each month, while the green and blue portions of the same bar represent the number of links existing in the given snapshot that are not present in the first month. In particular, the green portion of the bar represents the new links coming from old pages, i.e., pages that made their first appearance in the first snapshot, while the blue portion corresponds to new links coming from new pages, i.e., pages that did not exist in the first month. Figure 9 shows the same results applying a different normalization: the total number of links in each snapshot is normalized to one. Once again, the results we get are aligned to the ones reported in [9]: we observe that our data collection is characterized by a much more dynamic behavior in the evolution of the link structure rather than in the page dynamics. We notice that only 48% of the links in the first snapshot do exist in the second one, and only 25% of the first month links are present in the last crawl, that is, are still available after one year.

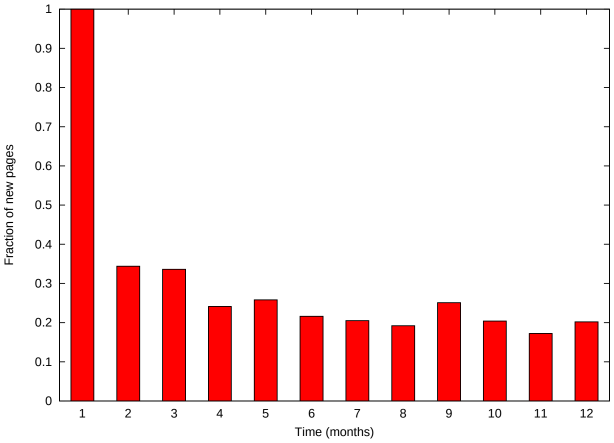

Figure 4: Fraction of new pages between consecutive snapshots.

6 Conclusions

In this paper we have presented a first study of a new large scale time-aware Web graph composed by twelve 100M pages monthly snapshots of the .uk domain. We have performed a set of assessment experiments that show that the graph is actually reliable.

We have studied some aspects of the temporal evolution of the dataset, performing an analysis that was inspired by [9]. Our findings seem to validate the results obtained in [9], as we still observe that existing pages are being removed by the Web and replaced by new pages at a very rapid rate. The evolution of the link structure is even faster than the one of the pages. We believe this similarity in the results to be a sign of the good quality of our data collection.

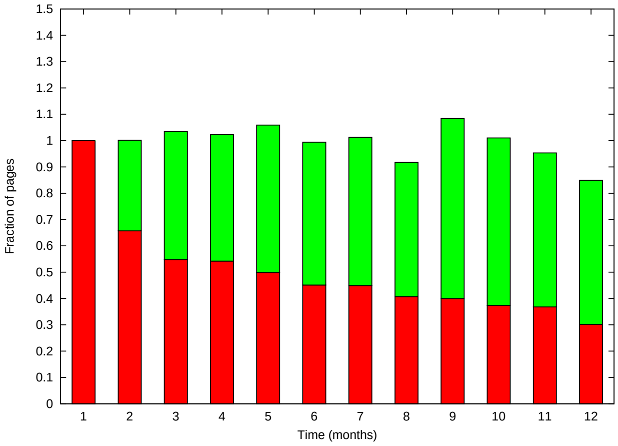

Figure 5: Fraction of pages from the first crawl still existing after n months(red bars) and new pages(green bars).

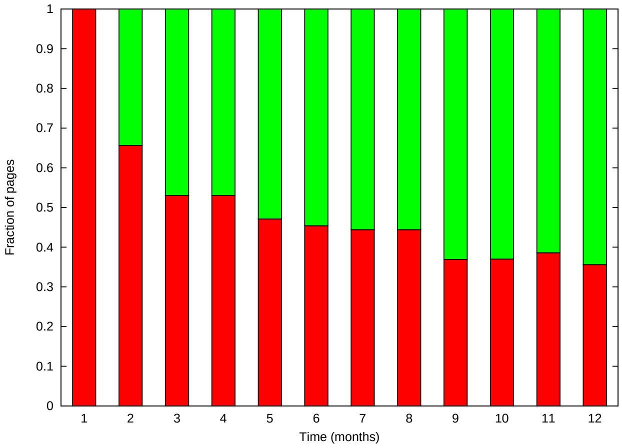

Figure 6: Normalized fraction of pages from the first crawl still existing after n months(red bars) and new pages(green bars).

Investigating how the content of Web pages changes over time is crucial to figure out how the Web is evolving. We plan to complete our study of the dynamics of the .uk domain by extensively analyzing the temporal evolution of the content of Web pages. The outcomes of such a study might be used in the attempt of modeling the age or the ”freshness” of Web pages.

Future work will also focus on how to deal with dynamic pages, which make particularly challenging the problem of identifying URLs in different snapshots corresponding to same Web page. We plan to develop a reasonable alignment

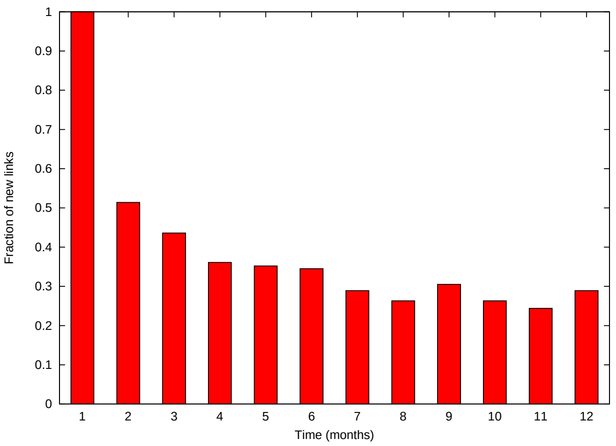

Figure 7: Fraction of new edges between consecutive snapshots.

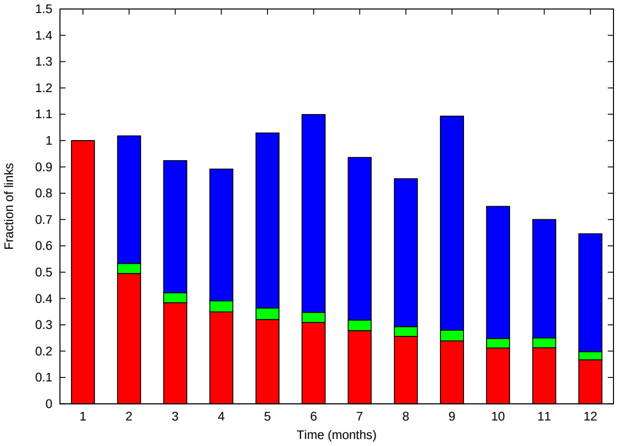

Figure 8: Fraction of edges from the first crawl still existing after n months (red bars), new edges from nodes existing in the first month (green bars), and new edges from new nodes (blue bars).

technique for dynamics URLs.

Acknowledgements

We are really thankful to Ricardo Baeza-Yates, Carlos Castillo, Aristides Gionis and Stefano Leonardi for several helpful discussions.

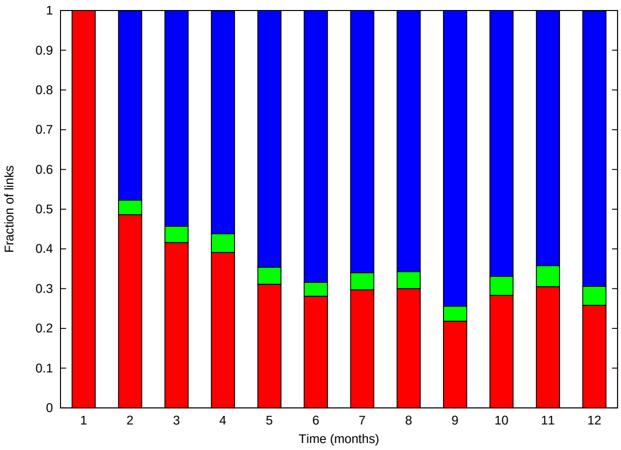

Figure 9: Normalized fraction of edges from the first crawl still existing after n months (red bars), new edges from nodes existing in the first month (green bars), and new edges from new nodes (blue bars).

References

[1] P. Boldi, B. Codenotti, M. Santini, and S. Vigna. Ubicrawler: A scalable fully distributed web crawler. *Software: Practice & Experience*, 34(8):711–726, 2004.

[2] P. Boldi, M. Santini, and S. Vigna. A large time-aware graph. *SIGIR Forum*, 42(1), 2008.

[3] P. Boldi and S. Vigna. The WebGraph framework I: Compression techniques. In *Proc. of the Thirteenth International World Wide Web Conference*, pages 595–601, Manhattan, USA, 2004. ACM Press.

[4] B. E. Brewington and G. Cybenko. Keeping up with the changing web. *Computer*, 33(5):52–58, 2000.

[5] J. Cho and H. Garcia-Molina. Estimating frequency of change. *ACM Trans. Internet Technol.*, 3(3):256–290, 2003.

[6] D. Fetterly, M. Manasse, M. Najork, and J. Wiener. A large-scale study of the evolution of web pages. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 669–678, New York, NY, USA, 2003. ACM.

[7] D. Gomes and M. J. Silva. Modelling information persistence on the web. In *ICWE '06: Proceedings of the 6th international conference on Web engineering*, pages 193–200, New York, NY, USA, 2006. ACM.

[8] W. Koehler. Web page change and persistence—a four-year longitudinal study. *J. Am. Soc. Inf. Sci. Technol.*, 53(2):162–171, 2002.

[9] A. Ntoulas, J. Cho, and C. Olston. What's new on the web?: the evolution of the web from a search engine perspective. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 1–12, New York, NY, USA, 2004. ACM.

[10] M. Toyoda and M. Kitsuregawa. What's really new on the web?: identifying new pages from a series of unstable web snapshots. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 233–241, New York, NY, USA, 2006. ACM.