Sub-document Timestamping of Web Documents

Yue Zhao and Claudia Hauff
Web Information Systems, Delft University of Technology, The Netherlands
{y.zhao-1,c.hauff@tudelft.nl}

ABSTRACT

Knowledge about a (Web) document's creation time has been shown to be an important factor in various *temporal information retrieval* settings. Commonly, it is assumed that such documents were created at a single point in time. While this assumption may hold for news articles and similar document types, it is a clear oversimplification for general Web documents. In this paper, we investigate to what extent (i) this simplifying assumption is violated for a corpus of Web documents, and, (ii) it is possible to accurately estimate the creation time of individual Web documents' components (so-called *sub-documents*).

Categories and Subject Descriptors: H.3.3 Information Storage and Retrieval: Information Search and Retrieval
Keywords: timestamping; sub-documents; Web archiving

1. INTRODUCTION

Accurately estimating at what point in time a (Web) document has originally been created is of importance for a number of applications, including the tracking of ideas over time, the detection of copied content, and temporal information retrieval (IR) — for some topics users might prefer to be served older Web documents, while for others users may prefer more recently created content.
Current research in Web-document based temporal IR usually considers either the documents' creation timestamp (i.e. when the document first appeared on the Web) or the extracted content timestamps (i.e. which time periods the document contains information about) as a raw signal to be included in retrieval models [2]. In this work we focus on the creation time of Web documents. Previous work, e.g. [9, 5, 8], has made the simplifying assumption that each Web document d_i has been created at one moment in time t_i and t_i can either be approximated by the first time the document (its URL) was crawled or by the first/oldest timestamp appearing in the document content. On the Web this is a highly unrealistic assumption — documents are constantly altered and updated, a classic example being blogs, which contain many different "sub-documents" (blog entries) created at different points in time. While the different sub-documents of a blog page may be easy to timestamp, for many other types of Web documents this is harder.
Thus, in this work, we aim to arrive at a first understanding of *sub-document timestamping*. Specifically, we empirically investigate the following two research themes:

RT1: To what extent do Web documents consist of sub-documents created at different times? What kind of documents contain two or more sub-documents? What is the timespan between the oldest and most recent sub-document of a document?

RT2: To what extent are we able to classify each sub-document as either having been created within the past month (relative to the document crawl time), within the past year or more than m years ago? What document features are most useful in the classification? Which type of sub-documents can we most accurately identify?

We investigate a subset of documents from the ClueWeb12 corpus[1] and date each document's paragraphs (a paragraph is a sub-document) individually based on historic Web crawl data collected from the Internet Archive[2] (IA).

Having dated all sub-documents, we first analyse this corpus of sub-documents before turning towards estimating the creation time of each sub-document with a standard machine learning pipeline.
We find that **two thirds** of the investigated Web documents (66.5%) do indeed contain sub-documents created at different points in time. More importantly, we also find a large gap between the oldest and most recently created sub-document (**1052 days on average**), indicating that relying on a single creation timestamp per document provides at best a very distorted picture of the true creation times. Classifying sub-documents according to their creation time using only sub-document internal features is possible with more than 66% of instances correctly classified.

2. RELATED WORK

Document creation timestamps are used in different temporal IR settings, such as timeline construction [11, 5], improving retrieval relevance [10, 8] and the estimation of a document's focus time [7].
A few existing works aim to infer the creation timestamps of document. De Jong et al. [4] built temporal language mod-

[1] `http://www.lemurproject.org/clueweb12.php/`
[2] `https://archive.org/`

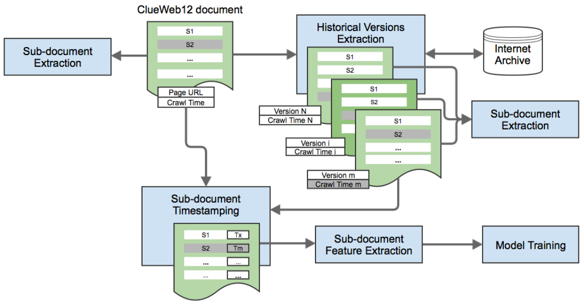

Figure 1: Overview of our processing pipeline for sub-document timestamping.

els from existing newspaper articles across a range of years and tagged non-timestamped articles based on the likelihood of being generated by a particular model. Kanhabua et al. [9] improve the temporal language model by using word interpolation, temporal entropy and external search statistics. They rely on documents recorded on the IA for their experiments, with the document's creation time being the first recorded crawl within the IA. Chambers et al. [3] use machine learning to infer documents' cration timestamps based on the temporal expressions by leveraging the MaxEnt model and additional time constraints. Ge et al. [6] propose an event-based time label propagation model by using the relationship between events and documents (exploiting the fact that news articles are often about events).

All these works infer a single creation time per document. In fact, most works [4, 9, 3, 6] rely on news corpora, which by design are rarely (or never) updated and usually contain an easily accessible creation timestamp. For *general Web documents*, there is little research work on inferring the sub-document creation timestamps. We attempt to fill this gap with our work.

3. APPROACH

To investigate our research questions we require a set of Web documents for which to determine the sub-document creation times. Instead of randomly sampling Web documents, we rely on the $11,075$ relevant documents D_{rel} available for the ClueWeb12 corpus (topics 201-300), which consists of more than 700 million English Web documents and was crawled between 02/2012 and 05/2012. We thus investigate documents that are at least relevant to some information needs based on their textual content, avoiding Web spam documents and Web documents that contain very little to no text in the process.

Historical Versions Extraction In Fig. 1 we present an overview of our pipeline. For each document in D_{rel} (identified through its URL), we retrieve all available historic versions from the IA, which began archiving Web documents in 1996. 7118 of the documents in D_{rel} contain at least one historic version. We continue our processing with those documents only ($D_{rel}^{archived}$). On average we are able to identify 17 historic versions per document in $D_{rel}^{archived}$.

Sub-document Extraction In the second step we identify the different sub-documents of each document $d_i \in D_{rel}^{archived}$ as well as the sub-documents of d_i's m historic versions $Hist_i = \{d_i^{h_1}, d_i^{h_2}, ..., d_i^{h_m}\}$ where h_1 is the most recent archived version of d_i (most recent but older than d_i's crawl date) and h_m is the oldest available version. In order to split a Web document d_i into k sub-documents $d_i = \{s_{1,i}, s_{2,i}, .., s_{k,i}\}$, we parse d_i's HTML. A sub-document is then a fraction split by tags `<p>` or `<div>`, which contains at least 50 non-markup characters. We empirically found this process to be a simple but robust mechanism to identify sub-documents. The number of sub-documents identified are on average 39 per document (median 21).

Sub-document Timestamping Let $Hist_i^{subdocs}$ be the set of all sub-documents created across *all* historic versions of document d_i. Then, for each sub-document $s_{i,j}$ of d_i we determine all matching (using approximate string matching) elements in $Hist_i^{subdocs}$ and assign to $s_{i,j}$ the creation timestamp of the oldest historic sub-document we found.

F1	Starting position of $s_{k,i}$ within d_i
F2	Number of terms in $s_{k,i}$
F3	Relative length of $s_{k,i}$: $\frac{\textit{length of } s_{k,i}}{\textit{length of } d_i}$
F4	Character distance between last position of $s_{k-1,i}$ and starting position of $s_{k,i}$
F5	Character distance between last position of $s_{k,i}$ and starting position of $s_{k+1,i}$
F6	Number of sentences in $s_{k,i}$
F7	Number of terms in the longest sentence in $s_{k,i}$
F8	Number of terms in the shortest sentence in $s_{k,i}$
F9	Average sentence length in $s_{k,i}$
F10	Number of temporal expressions in $s_{k,i}$
F11	Number of temporal expressions appearing before $s_{k,i}$
F12	Number of *Dates* in $s_{k,i}$
F13	Number of *Durations* in $s_{k,i}$
F14	Number of *Times* in $s_{k,i}$
F15	Number of *Sets* in $s_{k,i}$
F16	Difference in days between 1/1/1996 and the earliest temporal expression in $s_{k,i}$
F17	Difference in days between 1/1/1996 and the most recent temporal expression in $s_{k,i}$
F18	Difference in days between 1/1/1996 and the temporal expression in $s_{k,i}$ being closest to d_i's crawl time
F19	Difference in days between the earliest and most recent temporal expressions in $s_{k,i}$
F20	Average number of characters between the appearance of temporal expressions in $s_{k,i}$
F21	Longest character distance between the appearance of temporal expressions in $s_{k,i}$

Table 1: Features derived for sub-document $s_{k,i} \in d_i$. All features are based on the non-markup content.

Model Training Having identified for each sub-document its creation time, we now derive a set of 21 features in order to investigate **RQ2**. We restrict ourselves to document-internal features only.

The features are listed in Tab. 1. All features are based on the non-markup content extracted for a particular sub-document. While features F1 to F9 gather basic paragraph and sentence statistics, features F10 to F21 are based on the temporal expressions (TEs) we extract from a sub-document[3]. TEs can be classified into four different categories, depending on the specificity of the information: [F12] *Date* (e.g. *Feb. 18, 2015*), [F13] *Duration* (e.g. *from 1996 to 2012*), [F14] *Time* (e.g. *1pm*) and [F15] *Set* (e.g. *every weekend*). Since the focus of our work is an exploratory analysis of sub-document timestamping, we chose an established classifier with fixed parameter settings (Random Forest [1] with 5 features per tree and 100 trees in total) instead of experimenting with different algorithms and configurations.

We train & test the classifier on the 277K pairs of (*sub-document, sub-document creation timestamp*). We distinguish 5 classes and annotate each pair accordingly depending on the difference between a sub-document $s_{k,i}$'s creation time and the

[3] TEs are extracted with the SUTime tagger: `http://nlp.stanford.edu/software/sutime.shtml`.

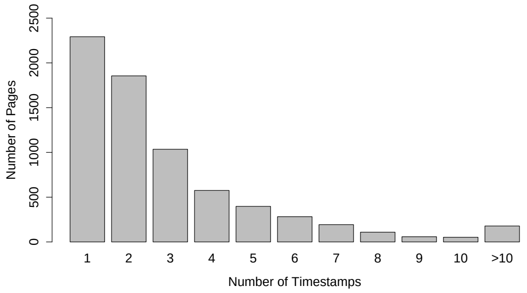

Figure 2: Overview of the number of documents containing content created at different points in time.

page crawl time of d_i[4]. We use the following five intervals: $A = [0, 20.5], B = (20.5, 311.5], C = (311.5, 973.5], D = (973.5, 2183.5]$ and $E = (2183.5, \infty)$. That is, class A contains those sub-documents created within the first 20 days of the page crawl time, while class E contains those sub-documents created more than 6 years before the page was actually crawled. We chose these interval settings to create a balanced data-set: each class has ∼55K instances.

In a second set of experiments we consider a subset of all instances, namely those 120K in which each sub-document contains at least one TE, as we aim to investigate the effect TEs have on the accuracy of the classification.

We employed the classifier to predict into which class a particular sub-document falls in a 10-fold cross-validation setup.

4. RESULTS

Sub-document timestamps.

Let us first consider **RT1** and the question to what extent sub-document timestamping is actually an issue on the Web. In Fig. 2 we plot the number of documents within $D_{rel}^{archived}$ and the number of *different* timestamps we assigned to their respective sub-documents. Overall, 62.5% of documents have between 2 and 8 creation timestamps; very few documents contain content created at eight or more different times (4%).

Since not only the number of different creation timestamps a document possesses, but also the time interval between the timestamps is important, in Fig. 3 we present the average difference (in days) between the oldest and most recent creation timestamp of a document, with the document set partitioned according to the total number of creation timestamps found in a document. For documents with two creation timestamps, the median difference is 400 days, i.e. 50% of those documents contain content created more than one year apart.

Considering these numbers we next investigate how much content is created at different points in time. For each document $d_i = \{s_{1,i}, s_{2,i}, .., s_{k,i}\} \in D_{rel}^{archived}$ with 2, 3 or 4 creation timestamps we determined what fraction of document content was created when. The results are shown in Fig. 4. Here, we consider all sub-documents (i.e. the non-markup text) of d_i as 100% of the content and compute what percentage of text was existing at each creation timestamp. This is a simplification of how Web documents are maintained (content might also be updated, deleted and added again over time). However, since we use the content of d_i as our starting point, we are only interested in the time a particular sub-document of d_i was first created. The graph shows that most content is created initially — for documents with 2 creation timestamps, on average 78% of the content is available after the first version of the document. For documents with 3 and 4 creation timestamps, 68% and 55% of content are created initially. Interestingly, the amount of content added in subsequent creation timestamps is roughly the same.

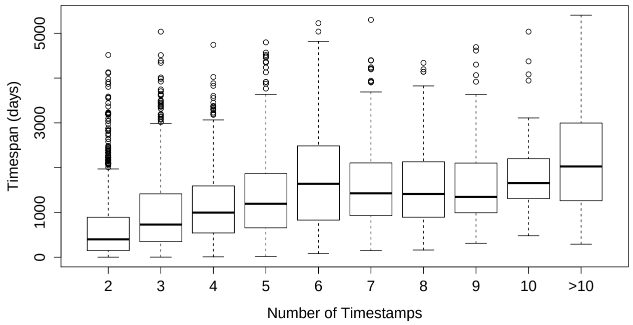

Figure 3: The document set $D_{rel}^{archived}$ is partitioned according to the number of creation timestamps (documents with a single creation timestamp are ignored). Shown is the difference (in days) between the oldest and most recent creation timestamp.

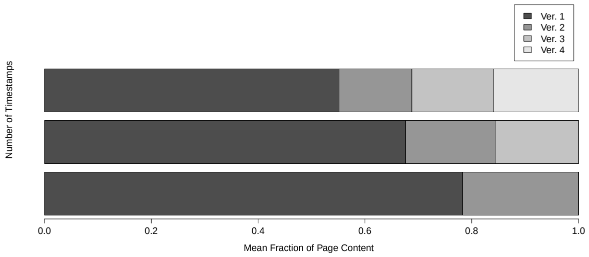

Figure 4: The document set $D_{rel}^{archived}$ is partitioned according to the number of creation timestamps (documents with a single creation timestamp are ignored). A bar shows the mean fraction of content available at each creation timestamp for documents with 2, 3 and 4 creation timestamps. *Ver. 1* indicates the content created at the oldest timestamp, *Ver. 2* the content created at the second oldest timestamp and so on.

Finally, we consider whether or not different information needs (topics) attract different kinds of documents, i.e. documents with few or many creation timestamps. Fig. 5 shows the distribution of documents with differing creation times for the 25 ClueWeb12 TREC adhoc topics with the largest number of relevant documents (the median number of relevant documents is 126). The results show that for most topics a relatively large percentage of relevant documents contain two or more creation timestamps. If we were able to predict what type of topics favour what kind of documents (a single creation time vs. several) we could employ these

[4] We assume that in practice a page's crawl time is usually available (as is the case for the ClueWeb12 corpus)

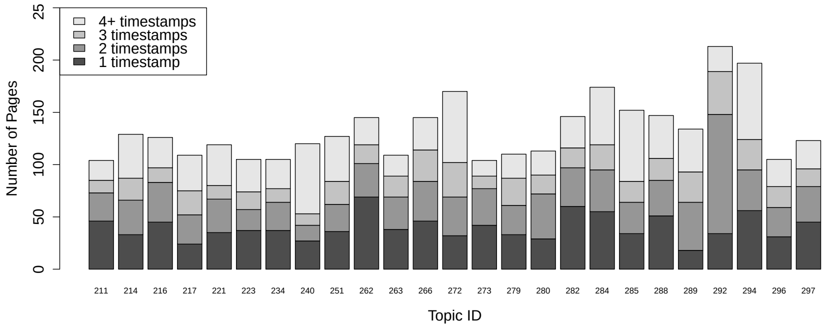

Figure 5: Overview of the relevant documents per TREC topic and the amount of creation timestamps.

	#Instances	Method	Misclassified	F-Measure / Class A	B	C	D	E
Entire Data Set	277,973	RF	33.73%	0.68	0.61	0.60	0.64	0.76
Data Set with TEs only	120,620	RF	33.12%	0.69	0.59	0.58	0.63	0.79
Data Set with TEs only	120,620	BL: earliest TE	60.90%	0.42	0.28	0.21	0.36	0.56
Data Set with TEs only	120,620	BL: latest TE	63.76%	0.33	0.29	0.19	0.36	0.50

Table 2: Effectiveness of our sub-document timestamp classification pipeline. *RF* refers to the Random Forest setup, while *BL* indicates the baseline, using a single feature only (oldest/most recent temporal expression appearing in the sub-document).

creation time-based signals in a retrieval ranking function (a direction of future work).

Predicting Sub-document Timestamps.

Our vision is to eventually develop techniques that are reliably able to tag any Web page's sub-documents with an accurate estimate of their creation time. To answer the questions raised in **RT2**, we consider the results of the creation timestamp classification experiments in Tab. 2. The Random Forest (RF) classifier classifies ∼65% of the instances correctly, independent of the existence of TEs in a sub-document (rows 1 & 2). Instances of class E (i.e. those sub-documents created 6+ years before the page crawl time) can be classified with highest accuracy. We also present the results of two baselines for those instances that contain one or more TEs: using as single feature either the oldest or most recent TE for classification purposes only. About two thirds of the instances are not correctly classified showing that TEs alone are not sufficient in this setup and additional features (which on first sight may not always be pertinent to creation timestamps) are required.

5. CONCLUSIONS

Our work shows that sub-document timestamping is an issue which should be considered when employing document creation timestamps in IR applications. Not only the amount of documents containing content created at several points in time is significant, but also the interval between the changes is considerable.

One of the limitations of our work is the fact that we relied on the Internet Archive and its historic versions of a document to determine each sub-document's creation time. While this approach yields very precise results for documents archived often by the Internet Archive, for less well-archived documents the temporal resolution is limited[5]. For this reason we resorted to a classification setup with five classes instead of estimating the exact creation time.

In future work we will (i) investigate the impact of sub-document timestamps on retrieval applications, and (ii) experiment with document-external features to increase the classification accuracy.

6. REFERENCES

[1] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[2] R. Campos, G. Dias, A. M. Jorge, and A. Jatowt. Survey of temporal information retrieval and related applications. *ACM Computing Surveys (CSUR)*, 47(2):15, 2014.

[3] N. Chambers. Labeling documents with timestamps: Learning from their time expressions. In *ACL '12*, pages 98–106, 2012.

[4] F. de Jong, H. Rode, and D. Hiemstra. Temporal language models for the disclosure of historical text. Royal Netherlands Academy of Arts and Sciences, 2005.

[5] L. Döhling and U. Leser. Extracting and aggregating temporal events from text. In *WWW '14*, pages 839–844, 2014.

[6] T. Ge, B. Chang, S. Li, and Z. Sui. Event-based time label propagation for automatic dating of news articles. In *EMNLP '13*, pages 1–11, 2013.

[7] A. Jatowt, C.-M. Au Yeung, and K. Tanaka. Estimating document focus time. In *CIKM '13*, pages 2273–2278, 2013.

[8] R. Jones and F. Diaz. Temporal profiles of queries. *ACM Transactions on Information Systems*, 25(3):14, 2007.

[9] N. Kanhabua and K. Nørvåg. Using temporal language models for document dating. In *Machine Learning and Knowledge Discovery in Databases*, pages 738–741. 2009.

[10] X. Li and W. B. Croft. Time-based language models. In *CIKM '03*, pages 469–475. ACM, 2003.

[11] R. Swan and D. Jensen. Timemines: Constructing timelines with statistical models of word usage. In *KDD Workshop on Text Mining*, pages 73–80, 2000.

[5] Note though, that this has only a very limited effect on the number of creation timestamps. Correlating the number of records of documents with the number of creation timestamps found in them, yields $r = 0.37$.