Exploiting the Social and Semantic Web for guided Web Archiving*

Thomas Risse
L3S Research Center
Appelstr. 9a
Hannover, Germany
risse@L3S.de

Stefan Dietze
L3S Research Center
Appelstr. 9a
Hannover, Germany
dietze@L3S.de

Wim Peters
University of Sheffield
211 Portobello Street
Sheffield S1 4DP, UK
w.peters@dcs.shef.ac.uk

Katerina Doka
IMIS / RC "ATHENA"
Artemidos 6
Athens 15125, Greece
katerina@cslab.ece.ntua.gr

Yannis Stavrakas
IMIS / RC "ATHENA"
Artemidos 6
Athens 15125, Greece
yannis@imis.athena-innovation.gr

Pierre Senellart
Institut Télécom
Télécom ParisTech; CNRS LTCI
Paris, France
pierre.senellart@telecom-paristech.fr

ABSTRACT

The constantly growing amount of Web content and the success of the Social Web lead to increasing needs for Web archiving. These needs go beyond the pure preservation of Web pages. Web archives are turning into "community memories" that aim at building a better understanding of the public view on e.g. celebrities, court decisions and other events. Due to the size of the Web, the traditional "collect-all" strategy is in many cases not the best method to build Web archives. In this paper we present the ARCOMEM architecture that uses semantic information such as entities, topics, and events complemented with information from the social Web to guide a novel Web crawler. The resulting archives are automatically enriched with semantic meta-information to ease the access and allow retrieval based on conditions that involve high-level concepts.

Categories and Subject Descriptors

D.2.11 [**Software Architectures**]: Domain-specific architectures; H.3.6 [**Library Automation**]: Large text archives

General Terms

Web Archiving, Web Crawler, Architecture, Text Analysis, Social Web

1. INTRODUCTION

Given the ever increasing importance of the World Wide Web as a source of information, adequate *Web archiving* and *preservation* has become a cultural necessity in preserving knowledge. The report *Sustainable Economics for a Digital Planet* [4] states that "the first challenge for preservation arises when demand is diffuse or weakly articulated." This is especially the case for non-traditional digital publications, e.g., blogs, collaborative space, or digital lab books. The challenge with new forms of publications is that there can be a lack of alignment between what institutions see as worth preserving, what the owners see as of current value, and the incentive to preserve together with the rapidness at which decisions have to be made. For ephemeral publications such as the Web, this misalignment often results in irreparable loss. Given the deluge of digital information created and this situation of uncertainty, a first necessary step is to be able to respond quickly, even if in a preliminary fashion, by the timely creation of archives, with minimum overhead enabling more costly preservation actions further down the line. This is the challenge that the ARCOMEM[1] project is addressing.

In addition to the "common" challenges of digital preservation, such as media decay, technological obsolescence, authenticity and integrity issues, web preservation has to deal with the sheer size and ever-increasing growth and change rate of Web data. Ntoulas et al. [16] showed that the web is growing by more than 8% per week and that after one year 40% of the pages are still accessible while 60% of the pages are new or changed. This is to be contracted with the fact that, according to Gomes at al. [10], only approximately 40 Web archiving initiatives are active which involve only about 270 people worldwide. Hence, selection of content sources becomes a crucial and challenging task for archival organizations.

A pivotal factor for enabling next-generation Web archives is crawling. Crawlers are complex programs that nevertheless implement a simple process: follow links and retrieve Web pages. In the ARCOMEM approach, however, crawling is much more complex, as it is enriched with functionality deal-

*This work is partly funded by the European Commission under ARCOMEM (ICT 270239).

[1]ARCOMEM – From Collect-All ARchives to COmmunity MEMories, `http://www.arcomem.eu/`

ing with novel requirements. Instead of following a "collect-all" strategy, archival organizations are trying to build *community memories* that reflect the diversity of information people are interested in. Community memories largely revolve around *events* and the *entities* related to them such as persons, organizations, and locations. These may be unique events such as the first landing on the moon or a natural disaster, or regularly occurring events such as elections or TV serials. Thus, entities and events are natural candidates for focusing new types of content acquisition processes in preservation as well as for archive enrichment.

Current Web crawler technology is mainly inspired or based on crawlers for search engines. Therefore they have limited or no notion of topics, entities, events, or the Social Web context. In this article we want to present the architecture of a new kind of Web crawler which addresses the special needs of Web archiving organizations. This new crawler generation will analyze and mine the rich social tapestry of the Social Web to find clues for deciding what should be preserved (based on its reflection in the Social Web), to contextualize content within digital archives based on their Social Web context, and determine how to best preserve this context. Contextualization based on the Social Web will be complemented by exploring topic-centered, event-centered, and entity-centered processes for content appraisal and acquisition, as well as for rich preservation.

The crawler architecture we propose in this paper is the basis for the current implementation activities in the ARCOMEM project. It should be noted that the system is only partially implemented yet; we are therefore not able to present any evaluation of the new system.

The remaining of the paper is structured as follows. The next section will present two example use cases and the derived requirements for the Web crawlers. Section 3 will give an overview about the overall architecture and the different processing phases. Section 4 describes the ARCOMEM data management approach for handling content and meta information. More details about the content and Social Web analysis as well as crawler guidance will be presented in Section 5. We discuss the state of the art in web archiving and related fields in Section 6. Finally, Section 7 gives conclusions and an outlook to future work.

2. USE CASES AND REQUIREMENTS

In order to develop a Web crawler which addresses the special needs of Web archiving organizations ARCOMEM follows a strong user-driven approach. Groups of potential users were actively involved in the design of the ARCOMEM platform, and later in the evaluation of the tools. The evaluation and verification of the ARCOMEM system from an early phase aimed at a better understanding of the requirements and adaptation of the functionality to the user needs.

To this end, two application scenarios have been selected, in order to illustrate and test in a variety of real-life settings the tools developed, and to provide feedback through mockups developed early in the project. The first application is driven by two major broadcasting companies, namely *Deutsche Welle* (DW) and *Südwestrundfunk* (SWR), and targets the event- and entity-aware enrichment of media-related Web archives based on the Social Web. The second application is driven by two European parliaments (the Greek and the Austrian parliament), and targets the effective creation of political archives based on the Social Web.

2.1 Broadcaster Use Case

Due to the increasing importance of the Web and Social Web, journalists will in the future no longer be able any more to only use reliable sources like news agencies, PR-material, or libraries. User-generated content will become another important information source. This shift in importance is also the case when own events of the broadcasters should be documented and their impact should be analyzed. In both cases it is important that the user-generated content stays accessible even if the original source disappears. Therefore, the management of digital content from a Social Web archive perspective is a key concern for broadcasting companies.

The main objective in the broadcaster scenario is to identify, preserve, interrelate, and eventually use multimedia content from the Social Web that is relevant to a specified topic, event, or entity. Two groups of users are involved: the *archivists* and the *journalists*. The archivists need support for selecting and archiving relevant content. Their job is to define and fine-tune the boundaries of the focused crawling until the results are satisfactory, at which point the results are stored into the archive. The journalists need to easily find relevant content for their stories/articles/shows, and then be able to follow the discussions and opinions on them.

As a concrete example, we consider the case of an annual popular rock festival called "Rock am Ring" that takes place in Germany and is covered by SWR. Journalists covering the festival would like to have access to relevant content from *blogs*, *social networks*, as well as *photo* and *video networks*. Information gathered from those sources is selected, processed, and organized so that questions such as the following can be answered:

- How did people talk about the event?
- How are opinions distributed in relation to demographic user data?
- Who are the most active Twitter users?
- Where did they come from?
- What did they talk about?
- What videos were most popular on Facebook?

2.2 Parliament Use Case

Parliament libraries provide Members of Parliament (MP) and their assistants, journalists, political analysts, and researchers information and documentation for parliamentary issues. Besides traditional publications, the Web and the Social Web play an increasingly important role as an information source since it provides important and crucial background information, like reactions to political events and comments made by the general public. It is in the interest of the parliaments to create a platform for preserving, managing, mining, and analyzing all the information provided in the social media.

Through ARCOMEM the Greek and Austrian parliaments aspire to transform their flat digital content archives to historical and community memories. In particular, one of the selected case studies concerns the Greek financial crisis. ARCOMEM opens the road for answering questions like:

- What is the public opinion on crucial social events?
- How has the public opinion on a key person evolved?
- Who are the opinion leaders?
- What is their impact and influence?

The parliament use case exhibits notable differences compared to the broadcaster use case. First, parliaments have multimedia archives with content partly produced by the parliamentary procedures. The focus is on associating this information with the events and entities involved, and subsequently enriching it with relevant user content from the Social Web. Second, crawls may last longer than in the broadcaster case. Political events may have a long aftermath, in contrast to news stories which are usually more temporally focused. Another difference is that a broad range of people use the parliament archives and may have varying requirements when retrieving information, making it difficult to cover everybody's needs.

2.3 Derived Requirements

The requirements for the ARCOMEM system have been compiled in close collaboration with the broadcaster and parliament users, and were based on the analysis of a number of use cases similar to those outlined above. The analysis phase identified a list of possible content sources for each use case, belonging to various social media categories (blogs, wikis, social networks, discussion groups, etc.), together with a number of attributes of interest for each source. Moreover, the functionality of the system was specified in detail. The requirements analysis led to the definition of the ARCOMEM system architecture, which is discussed extensively in the following sections.

3. APPROACH & ARCHITECTURE

The goal for the development of the ARCOMEM crawler architecture is to implement a socially aware and semantic-driven preservation model. This requires thorough analysis of the crawled Web page and its components. These components of a Web page are called *Web objects* and can be the title, a paragraph, an image or a video. Since a thorough analysis of all Web objects is time-consuming, the traditional way of Web crawling and archiving is no longer working. Therefore the ARCOMEM crawl principle is to start with a *semantically enhanced crawl specification* that extends traditional URL based seed lists with semantic information about entities, topics or events. This crawl specification is complemented by a small reference crawl to learn more about the crawl topic and intention of the archivist. The combination of the original crawl specification with the extracted information from the reference crawl is called the *intelligent crawl specification*. This specification, together with relatively simple semantic and social signals, is used to guide a broad crawl that is followed by a thorough analysis of the crawled content. Based on this analysis a semi-automatic selection of the content for the final archive is carried out.

The translation of these steps into the ARCOMEM system architecture foresees four processing levels: the *crawler level*, the *online processing level*, the *offline processing level*, and *dynamics analysis*, that revolve around the ARCOMEM database as depicted in Figure 1. The ARCOMEM database – consisting of an *object store* and a *knowledge base* – is the focal point for all components involved in crawling and content analysis. It stores all information from the crawl specification over the crawled content to the extracted knowledge. Therefore a scalable and efficient implementation together with a sophisticated data model is necessary (see Section 4). The different processing levels are described as follows:

3.1 Crawling Level

At this level, the system decides and fetches the relevant Web objects as these are initially defined by the archivists, and are later refined by both the archivists and the online processing modules. The crawling level includes, besides the traditional crawler and its decision modules, some important data cleaning, annotation, and extraction steps (we explain this in more detail in Section 5.5). The Web objects (i.e., the important data objects existing in a page, excluding ads, code, etc.) are stored in the ARCOMEM database together with the raw downloaded content.

3.2 Online Processing Level

The online processing is tightly connected with the crawling level. At this level a number of semantic and social signals such as information about persons, locations, or social structure taken from the intelligent crawl specification are used to prioritize the crawler processing queue. Due to the near-real-time requirements, only time-efficient analysis can be performed, while complex analysis tasks are moved to the offline phase. The logical separation between the online processing level and the crawler level will allow the extension of existing crawlers at least with some functionalities of the ARCOMEM technology.

3.3 Offline Processing Level

At this level, most of the basic processing over the data takes place. The offline, fully-featured, versions of the entity, topics, opinions, and events analysis (ETOE analysis) and the analysis of the social contents operate over the cleansed data from the crawl that are stored in the ARCOMEM database. These processing tools perform linguistic, machine learning and NLP methods in order to provide a rich set of metadata annotations that are interlinked with the original data. The respective annotations are stored back in the ARCOMEM database and are available for further processing and information mining. After all the relevant processing has taken place, the Web pages to be archived and preserved are selected in a semi-automatic way. Finally, the selected original pages are transferred to the Web archive (in the form of WARC files, s. Section 3.5).

3.4 Dynamics Analysis Level

Finally, a more advanced processing step takes places. It operates on collections of Web objects that have been collected over time in order to register the evolution of various aspects

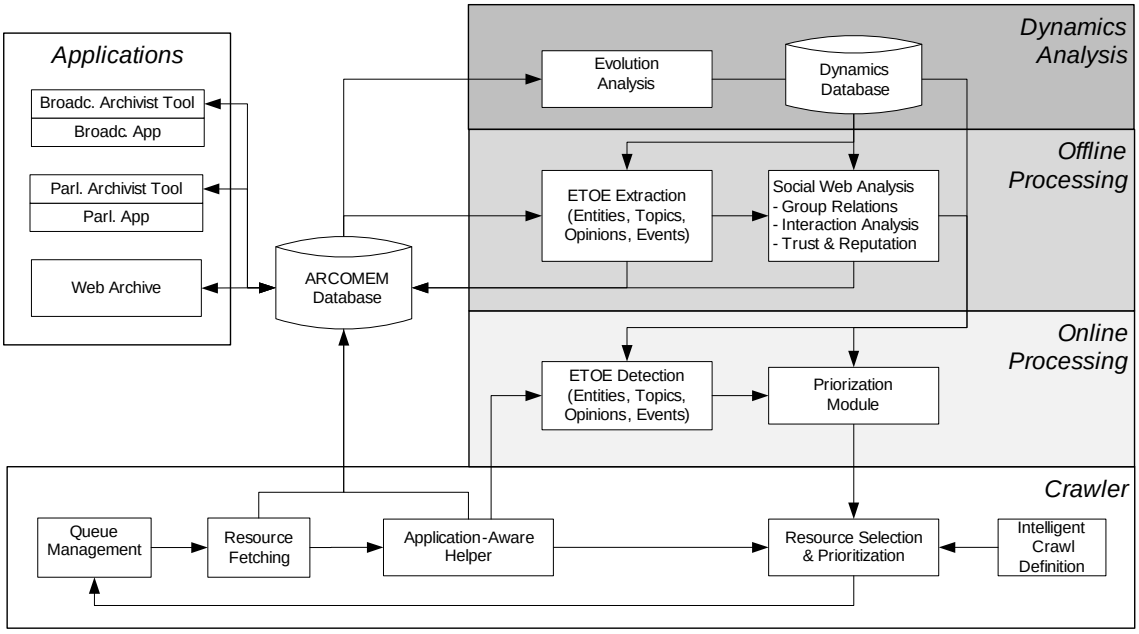

Figure 1: Overall Architecture

identified by the ETOE and Web analysis components. As such, it produces aggregate results that pertain to a group archive of objects, rather than to particular instances.

3.5 Applications

We implement customized methods to interact with the ARCOMEM crawler and ARCOMEM database, for example to satisfy the use cases of Section 2. The *archivist tools* allow archivist to specify or modify crawl specifications and do the quality assurance. By explicitly marking certain pages as relevant to a crawl, the intelligent crawler can – even during a crawl campaign – learn more about the crawl intentions and crawl specification. This is especially important for long running crawls with broader topics. The intentions behind broader crawl topics are less precise and rather abstract, which leads to a semantically more generic crawl specification. Examples for this are the financial crisis or elections. In these cases, sub-topics, entities, and events are changing more often than during highly focused crawls and therefore require regular adaption of the crawl specification.

The archivist tool is also used to create the final Web archives. Based on a relevance analysis, a semi-automatic method proposes to the archivist relevant Web pages from the ARCOMEM database that should be preserved. The archivist always has the possibility to include or exclude pages from this selection. Finally, the selected content will be transferred to the WARC files for preservation.

The foreseen end-user applications allow users to search the archives by domain, time and keywords. Furthermore, browsing the archives via different facets like topics, events, and entities, and visualizing the sentiments of Social Web postings complement the end user application. However, the applications are not limited to the described examples. The ARCOMEM system is open to any kind of application that wants to use it.

4. ARCOMEM DATA MANAGEMENT

The manipulation of the data and metadata produced and used throughout all architectural levels is an important task that needs to comply with the functional as well as the technical requirements of a semantically charged, social web crawler. This section defines the data model by identifying the important concepts that need to be preserved and describes the way to efficiently store and handle crawled content as well as derived annotations.

4.1 ARCOMEM Data Model

One of the first tasks to be addressed in ARCOMEM was the identification of the concepts that are relevant for knowledge capturing, crawling, and preservation. After a requirements analysis, a structured domain representation was created that reflects the informational needs of ARCOMEM.

The central concepts in this configuration are *InformationObject*, *InformationRealization* and *CrawlingConcept*. InformationObject is the class that subsumes all ARCOMEM's information types: Entities, Events, Opinions, and Topics. Multilingual instances of this class are classified according to the language they belong to. InformationRealization captures the concrete instantiations of these information objects in the form of multimedia web object such as texts and images. CrawlingConcept describes required aspects of the crawling workflow.

The creation of links to concepts from various de facto or officially established standardized vocabularies ensures a proper embedding of the ARCOMEM conceptual vocabulary in the wider semantic web context. Where possible, links with existing concepts in the linked open data cloud are increasingly being established.

Figure 2 illustrates the resulting data model in the form of a simplified UML notation in the form of classes and

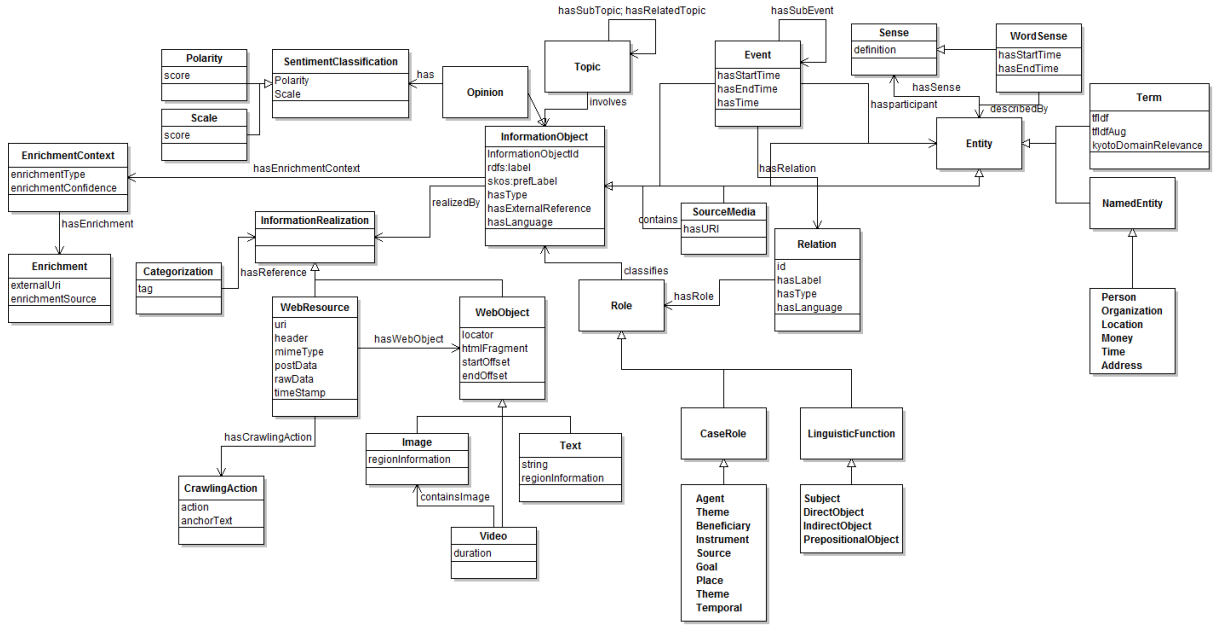

Figure 2: The ARCOMEM Data Model

their relations. Open arrows represent subClassOf relations, whereas closed arrows indicate any other type of relation. The ARCOMEM data model is represented in RDF[2] format. The serialization and population of this model in RDF enables structured access to this information through SPARQL queries[3].

4.2 ARCOMEM Database

The ARCOMEM database is a component that plays a central role in the platform. Its task is to provide storing, indexing, and retrieving mechanisms for all data produced and utilized by the rest of the architectural components. As such, it is expected to store, serve, and update different kinds of data: (a) binary data, in the form of Web objects, which represent the original content collected by the crawler; and (b) semi-structured data, in the form of RDF triples, which serve as Web object annotations and are primarily used by the ETOE and Social Web analysis, the dynamics analysis, as well as the applications.

The design and implementation of the ARCOMEM database is also dictated by non-functional requirements. The sheer volume of information available on the Internet combined with the requirement of our system to capture multiple versions of Web objects over time creates enormous demands for storage as well as memory. Moreover, as some decisions are made at runtime (e.g., during the online processing), queries need to be resolved in near-real-time. Even for complex analytics tasks, high throughput is important since they may trigger a large number of queries and thus take hours or even days to complete.

[2]The *Resource Description Framework (RDF)* is the W3C standard for the conceptual description and modeling of information that is implemented in web resources. It relies on statements expressed in the form of subject-predicate-object *triples*, denoted as $\{S, P, O\}$.
`http://www.w3.org/RDF/`

[3]http://www.gate.ac.uk/ns/ontologies/arcomem-data-model.rdf

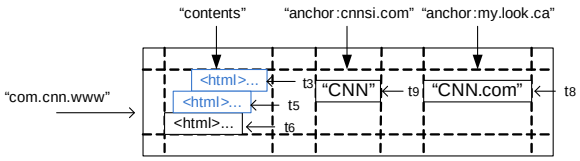

Figure 3: Sample content of the Object Store

To cover the functional requirements, we have designed and implemented a storage module consisting of two components: the *object store* and the *knowledge base*. Both of them rely on distributed solutions that combine the MapReduce [9] programming model and NoSQL databases to cover the non-functional requirements. MapReduce is an ideal paradigm for harnessing scale-out architectures to build huge indices on distributed storage, while NoSQL databases, based on shared-nothing architectures, offer scalability and availability at low cost.

The ARCOMEM Object Store. The object store is responsible for storing and serving the raw content harvested by the crawler. It relies on HBase [1], a NoSQL database written in Java and modeled after Google's BigTable [6]. Essentially, each table is a sparse map storing values in cells defined by a combination of a row and a column key. In order to provide high availability of the data, HBase keeps its data in a distributed filesystem called HDFS [17]. This approach also diminishes the impact of a node failure on the overall performance.

A sample row of the Object Store is shown in Figure 3. The URL of the fetched Web object, whether it is text, image or video, will serve as the row key (`www.cnn.com` in our case),

while the actual binary content will be stored under "content". Extra attributes can be stored in other table columns. Moreover, it is possible to store many timestamped versions of a fetched Web object under the same row key. This is important, since the crawling procedure is likely to discover altered version of already stored Web objects in different crawls. It is also important to note that the object store allows a posteriori addition of attributes as well, allowing the data model to be enriched on demand.

Querying is performed using either the HBase Get API, which retrieves a particular Web object by its URL, or the HBase Scanner API, which sequentially reads a range of URLs. In both cases the user can control which version of the datum is to be retrieved. The object store can be queried from a client over the network or using distributed MapReduce computations. For this purpose, Java classes as well as a RESTful interface are provided.

The ARCOMEM Knowledge Base. The task of the knowledge base is to handle the data that derives from the annotation of the Web objects, as performed by the online as well as the offline processing modules. Such annotations are described using the *ARCOMEM data model* (see above) that interlinks ETOEs and point to actual content residing in the Object Store. Thus, the problem translates to building an efficient processing engine that appropriately indexes and stores RDF triples and offers SPARQL querying capabilities, while maintaining the scalability and high-performance characteristics.

Existing solutions fail to simultaneously offer both query expressiveness and scalability/performance. Thus, we have designed a distributed triple index solution, following a hybrid approach that combines the power and expressiveness of an RDF engine with the performance and scalability of NoSQL.

More specifically, we have implemented an enhanced version of the centralized Hexastore indexing scheme [20] over HBase. Hexastore is an RDF store that creates 6 indices (for all possible permutations of subject, predicate and object $\{S, P, O\}$ – SPO, PSO, POS, OPS, SOP, OSP) in main memory, thus offering the ability to retrieve any triple pattern with minimal cost. Taking this one step further, our distributed knowledge base creates three indices ultimately stored in an HBase table. These indices correspond to three combinations of S, P and O, namely SP_O, PO_S and OS_P. The knowledge base is able to provide native SPARQL query functionality, with joins being executed as MapReduce jobs. Figure 4 pictorially presents the knowledge base architecture. Initial experiments prove the ability of the knowledge base to scale to billions of triples and to handle concurrent user requests, achieving fast response times.

5. ANALYSIS FOR CRAWL GUIDANCE AND ENRICHMENT

We now describe in more detail the analyses that are performed at all levels of the ARCOMEM architecture in order to guide the crawl towards the content given in the intelligent crawl specification, and in order to enrich the crawled

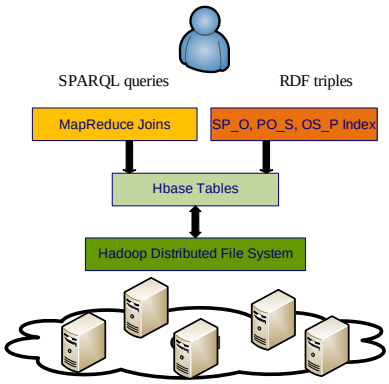

Figure 4: The Knowledge Base architecture

content with useful information. We discuss over content analysis, analysis of the social Web, dynamics analysis, data enrichment, and crawler guidance itself.

5.1 Content Analysis

In order for archivists to judge the content of a particular Web document, and decide whether it should be archived, they need to be able to assess this content. The aim of this module is the extraction and detection of informational elements called ETOEs (Entities, Topics, Opinions, and Events) from Web pages (s. Section 3. The ETOE extraction takes place in the offline phase and processes a collection of Web pages. The results of the offline ETOE extractions are used to (1) get a better understanding of the crawl specification and (2) populate the ARCOMEM knowledge base with structured data about ETOEs and their occurrence in Web objects. In the online phase, single documents will be analyzed to determine their relevance to the crawl specification.

A crawl campaign is described by a crawl specification given by the archivist. This specification consists of, in addition to other parameters, a search string where the archivist specifies in their own words the semantic focus of the crawl campaign. The search string is a combination of entities, topics, and events, plus free terms. Since it will not always be possible to literally match the search string with the content of a Web page, it is important to learn from an initial set of pages how the search string will be represented on real pages. This analysis will be done in the offline phase since it requires a collection of Web pages and is computationally more expensive. The result of this analysis is used in the online phase to derive the relevance of a page with respect to the crawl specification.

The extraction of ETOEs from Web pages is performed by robust and adaptable processing tools developed within the GATE architecture [8][4]. GATE is a framework for language engineering applications, which supports efficient and robust text processing. GATE uses NLP based techniques to assist the knowledge acquisition process, applying automated linguistic analysis to create text annotations and conceptual knowledge from textual resources. Documents are analyzed by linguistic pre-processing (tokenisation, language detection, sentence splitting, part of speech tagging, morpholog-

[4] `http://gate.ac.uk/`

ical analysis, and verb and noun phrase chunking). This is then followed by the extraction of both named entities using ANNIE [8] and term candidates using TermRaider[5]. Both tools use rule-based heuristics, while TermRaider applies statistical measures in order to determine termhood.

Linguistic analysis and extracted terminology are then used to identify events. Events such as crises, downgradings, and protests express relations between entities, which our module identifies both by statistical means and lexico-syntactic patterns such as linguistic predications expressed by verbal constructs.

Besides text content, pictures and videos will also be used to support and complement the entity extraction and detection process. If suitable training images are provided, classification and automatic annotation techniques can be used to predict entities within images, as well as topics and events depicted by the image as a whole. The training and data collection for these classifiers is an offline process that is itself guided by the crawl specification.

In addition to content acquisition on the basis of the initial archivist's crawl specification (generally consisting of a search string), the extracted ETOEs are used to enrich the crawled documents with meta-information which can later be used for retrieval or for further analysis tasks of the applications. By using unique URIs for entities and events, it will also be possible to search for the same entity across archives.

The result of the content acquisition is a structured set of extracted ETOEs, which is stored in a knowledge base according to the data model described in Section 4.1.

In the online phase, a newly crawled Web page will be analyzed to detect the ETOEs that were extracted in the offline phase. Afterwards, the coverage of the page will be compared with the crawl specification. This information is used by the intelligent crawler to prioritise the extracted URLs. A high overlap between the ETOEs on the newly crawled page and the ETOEs in the crawl specification indicate a highly relevant page and thus a high priority for the page and its URLs.

Statistical methods will be used to predict events from temporal anomalies or clusters of entities in a document corpus. This kind of event detection may also be applied to images or videos directly by computing their respective similarity. Visual depictions of salient events often have a high degree of similarity. The prototypical visual signatures of an event learnt in the offline analysis phase can be applied in the online phase to help predict relevance. Differences in opinion may also be related to this, as images in documents that portray the same opinion tend to be clustered, whereas images in documents portraying different opinions tend to be more dissimilar. However, in most cases the concept of image similarity needs to be much more semantically motivated than just considering things like the difference in pixel values. Again, the prototype signatures calculated in the offline phase can be used to predict relevance in the online phase; but perhaps more importantly, they might also be used to help ensure that the archive contains a diverse set of documents with respect to their opinions. Finally, images can potentially be very useful in aiding the disambiguation of entities extracted from the textual content of the document. In particular, given a set of different hypotheses from textual analysis, the image content can potentially be used to give weight to the most likely hypothesis.

[5] `http://gate.ac.uk/projects/neon/termraider.html`

5.2 Social Web Analysis

The aim of the Social Web analysis is to leverage the Social Web to contextualize content and information to be preserved, and to support the crawler guidance. In social networks users are discussing and reflecting about all kinds of topics, events and persons. By doing, so they regularly post links to other relevant Web pages or Social Web content. As these links are recommendations of individuals in the context of their social online activities they are highly relevant for preservation. However, since users are unknown and anonymous it is necessary to derive their reputation and trustworthiness in the social community during the Social Web analysis. This is for example a helpful method to identify spammers and consequently reduce the crawling priority for such users.

The Social Web is not only a valuable source of information. It also raises challenges for the crawler as the public access to the content is restricted by site owners, for example by limiting the access frequency. To get the most interesting content in such cases general interaction patterns of users in social networks need to be identified. The interaction analysis process discovers interaction patterns and cultural dynamics in social networks, where information propagation through the network is monitored, e.g., the public reaction to a particular event or the interaction between (groups of) users are analyzed. This information can be used, depending on the crawl intention, to increase or decrease to priority of these persons. Such analysis is part of current research in the ARCOMEM project.

The results of the interaction analysis can also be leveraged in the contextualization process to further enrich the web objects, e.g., if the object is tweeted by many nature experts it may be a good candidate for nature topics. Furthermore, the similarity and overlap between the provided objects and objects already seen before is established in order to interlink those that are discussing the same event, activity, or entity, improving the contextualization of the involved web objects.

5.3 Dynamics Analysis

The aim of the dynamics analysis module is to capture dynamics in entities, events, opinions, and terms across several crawls and therefore over time. For the Social Web content analysis the ETOE extraction module will especially benefit from the term and entity evolution detection. A major property that can be observed in Social Web networks is their high dynamicity and communication frequency. As a result the messages exchanged are rather short and the authors are trying the "optimize" their entries by deriving new words or acronyms. This is especially the case for microblogging sites like Twitter but it can also be observed in other systems. Some of these derived terms become standards while other might disappear again after a short time.

The written language in these environments is evolving with a higher rate than has been observed in traditional media. The content analysis module needs to be aware of these evolutions in order to better support the decision making process in the online phase. But capturing the dynamics is not only important for the crawling, since it provides valuable information for the scientific community, having the ability to answer questions such as how language changes over time. It is, however, also valuable to the crawl owners. In the broadcaster use case, opinions are sought concerning the rock festival: here it can be of great importance to know how opinions about this festival have changed over time, and what events or changes may have contributed towards these opinion changes, e.g., fewer bands might decrease popularity.

As shown in the functional overview in Figure 1, the dynamics module is located within a separate higher layer to indicate that the required input is a set of crawls or one long spanning crawl. The analysis is triggered once a sufficient time span has been represented in the archive or by the Web objects, in order to increase the probability of finding evolutions. The time span required differs depending on which type of evolution is sought and which type of data is represented in the crawl. Social Web content and user-generated data tend to be more dynamic and fast changing than printed newspapers or static Web pages, and thus require a shorter time span.

When the dynamics analysis is triggered, it will access the ARCOMEM database to find document annotations representing entities, events, opinions, and terms in the documents. The evolution modules then focus on clustering terms, for example, in order to detect if there has been any evolution of the term. Evolutions are defined differently depending on their context; for terms it involves a term's usage, its meanings, and its mappings to other terms. For example, the evolution module should map the term "iPod" to the term "MP3 player", as well as explaining its meaning as a music playing device. From the term's usage, the module could then deduce that the first iPods were manipulated using buttons, while the later ones were by means of a touch-screen. The term evolution helps the content analysis process and crawler to identify new upcoming terms or acronyms that are used instead or in addition to well known terms. More details on term evolution can be found in [18, 19].

5.4 Data Enrichment and Consolidation

Because the dynamics analysis and the content analysis extract structured data from unstructured resources, such as text and images, the generated data is heterogeneous. This is due to the data being generated by different components and during independent processing cycles. For instance, during one particular cycle, the text analysis component might detect an entity from the term "Ireland", while during later cycles, entities based on the term "Republic of Ireland" or the German term "Irland" might be extracted. These would all be classified as entities of type *arco:Location* and correctly stored in the ARCOMEM data store as separate entities described according to the ARCOMEM RDF schema. Data enrichment and consolidation follows two aims: (a) enrich existing entities with related publicly available knowledge; and (b) identify data correlations such as the ones above by aligning ARCOMEM entities with reference datasets. Both (a) and (b) exploit publicly available data from the Linked Open Data cloud[6] which offers a vast amount of data of both domain-specific and domain-independent nature.

To achieve the described aims, the enrichment approach first identifies correlating enrichments from reference datasets which are associated with the respective entities and, secondly, uses these shared enrichments to identify correlating entities in the ARCOMEM knowledge base. In particular, the current enrichment approach uses DBpedia[7] and Freebase[8] as reference datasets, though it is envisaged to expand this approach with additional and more domain-specific datasets, e.g., event-specific ones. DBpedia and Freebase are particularly well-suited due to their vast size, the availability of disambiguation techniques which can utilise the variety of multilingual labels available in both datasets for individual data items and the level of inter-connectedness of both datasets, allowing the retrieval of a wealth of related information for particular items. We distinct *direct* as well as *indirect* correlation. To give an example for direct correlations based on our current enrichment implementation, for instance, the three entities mentioned above, all referencing to the same real-world entity, are each associated with the same enrichments to the respective Freebase (`http://www.freebase.com/view/en/ireland`) and DBpedia (`http://dbpedia.org/resource/Ireland`) entries. Therefore, correlated ARCOMEM entities (and hence, Web objects) can be clustered directly by identifying joint enrichments between individual entities.

In addition, the retrieved enrichments associate (interlink) the ARCOMEM data and Web objects with the vast knowledge, i.e., data graph, available in the LOD cloud, thus allowing to retrieve additional related information for particular ARCOMEM entities. For instance, the DBpedia RDF description of Ireland (`http://dbpedia.org/page/Ireland`) provides additional data, facts, and knowledge (for instance, a classification as island or country, geodata, the capital or population, a list of famous Irish people and similar information) in a structured, and therefore, machine-processable form. That knowledge is used to further enrich ARCOMEM entities and create a rich and well-interlinked *(RDF) graph of Web objects and related information*. Thus, we can perform additional clustering and correlation of entities (and hence, crawled Web resources) to uncover indirect relationships between Web resources related in one way or another. For instance, Web resources about topics such as *Dublin*[9], *James Joyce*[10] or the *IRA*[11], can be associated and correlated simply by analysing the DBpedia graph to identify correlations between existing enrichments. Our initial enrichment and clustering technique is already implemented and currently under evaluation. Initial experiments were able to retrieve a total of 226 enrichments (i.e., DBpedia and Freebase references) for a sample set of 1095 extracted entities. While in a large graph such as DBpedia, any node

[6] `http://lod-cloud.net/`
[7] `http://dbpedia.org/`
[8] `http://www.freebase.com/`
[9] Enriched with a reference to `http://dbpedia.org/resource/Dublin`, which directly links to `http://dbpedia.org/resource/Ireland`.
[10] `http://dbpedia.org/resource/James_Joyce`
[11] `http://dbpedia.org/resource/Irish_Republican_Army`

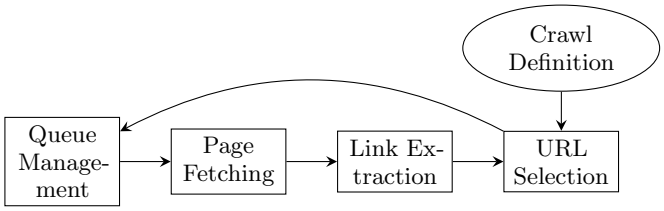

Figure 5: Traditional processing chain of a Web crawler

is connected with each other in some way, key research challenges are the investigation of appropriate graph navigation and analysis techniques to uncover indirect but meaningful relationships between ARCOMEM Web objects.

5.5 Crawler Guidance

Consider first the simplified view of the architecture of a traditional Web crawler depicted on Figure 5. In a traditional Web crawler, such as Heritrix [15], the archiving task is described using a crawl definition configuration file that specifies the list of seed URLs to start the crawl from, patterns that specify which URLs to include or exclude from the crawl, etc. At runtime, URLs are managed in a priority queue, which ensures optimal allocation of the bandwidth available to the crawler given the URLs of the frontier (URLs discovered but not yet crawled) and politeness constraints. Web pages are then fetched one after the other, links are extracted from these Web pages and the crawl definition is used to determine whether the URLs found on a Web page should be added to the frontier.

We expand on this architecture in the ARCOMEM project, adding new functionalities to the crawling system as shown in the bottom part of Figure 1. As already noted, the traditional crawl definition file is replaced by an intelligent crawl definition, which allows the specification of relevance scores and the referencing of the particular kinds of Web applications and ETOEs that define the scope of the archiving task. Queue management functions similarly as in a traditional architecture, but the page fetching module is replaced by some more elaborate *resource fetching* component that is able to retrieve resources that are not just accessible by a simple HTTP GET request (but by a succession of such requests, or by a POST request, or by the use of an API, a very common situation on Social Web networks), or individual Web objects inside a Web page (e.g., blog posts, individual comments, etc.). Each content item obtained by the crawler is stored in the ARCOMEM database for use by analysis modules and archivist tools.

After a resource (for instance a Web page) is fetched, an *application-aware helper* module is used in place of the usual link extraction function, in order to identify the Web application that is currently being crawled, decide on and categorize crawling actions (e.g., URL fetching, using an API) that can be performed on this particular Web application, and the kind of Web objects that can be extracted. This is a critical phase for using clues from the Social Web to crawl content, because, depending on the kind of Web application that is being crawled (traditional static Web site, Web forum managed by some content management system, wiki, social networking sites such as Twitter or Facebook), the kind of relevant crawling actions and Web objects to be extracted vary dramatically.

The crawling actions thus obtained, depending on their nature (embeds or other Web resources) are either directly forwarded to the selection component or sent for further analysis and ranking to the online analysis modules. Since crawling actions are more complex than in a traditional crawler, and since we want to prioritize the crawl in an intelligent manner, the URL selection module is replaced by a *resource selection & prioritization* module that makes use of the intelligent crawling definition and of the feedback from the online analysis modules to prioritize the crawl. This is the step where semantic analysis can make an impact on the guidance of the crawl: for example, if a topic relevant to the intelligent crawl specification is found in the anchor text of a link pointing to an external Web site, this link may be prioritized over other links on the page; if a Twitter user referenced on a Web page is detected to be part of a spammer's network, the system might decide not to follow any of the links encountered on the page; etc.

6. RELATED WORK

Since 1996, several projects have pursued Web archiving (e.g., [2]). The Heritrix crawler [15], jointly developed by several Scandinavian national libraries and the Internet Archive through the International Internet Preservation Consortium (IIPC)[12], is a mature and efficient tool for large-scale, archival-quality crawling.

The method of choice for memory institutions is client-side archiving based on crawling. This method is derived from search engine crawl, and has been evolved by the archiving community to achieve a greater completeness of capture and a reduction of temporal coherence of crawls. These two requirements follow from the fact that, for web archiving, crawlers are used to build collections and not only to index [13]. These issues were addressed in the European project LiWA (Living Web Archives)[13].

The task of crawl prioritization and focusing is the step in the crawl processing chain which combines the different analysis results and the crawl specification for filtering and ranking the URLs of a seed list. The filtering of URLs is necessary to avoid unrelated content in the archive. For content that is partly relevant, URLs need to be prioritised to focus the crawler tasks to crawl in order of relevancy. A number of strategies and therefore URL ordering metrics exist for this, such as breadth-first, back link count and PageRank. PageRank and breadth-first are good strategies to crawl "important" content on the web [3, 7], but since these generic approaches do not cover specific information needs, focused or topical crawls have been developed [5,14]. However, these approaches have only a vague notion of topicality and do not address event-based crawling.

A limited set of tools exist for accessing Web archives like NutchWAX and Wayback. NutchWAX [11] is based on

[12] `http://netpreserve.org/`
[13] `http://www.liwa-project.eu/`

Apache Nutch[14] - an open source web-search software project that supports URL and keyword based access. It adapts Nutch by searching against Web archives rather than crawling the Web. Wayback [12] is an open source implementation of the Internet Archive Wayback Machine. It allows browsing the history of a page or domain over time. Overall the possibilities to explore Web archives are limited to basic functionalities. The ARCOMEM enriched Web archives will allow accessing and browsing by a number of different facets. A step in this direction has already been made in the LivingKnowledge[15] project, where diversification of topics and opinions has been applied to Web archive content to improve search results.

7. CONCLUSIONS & FUTURE WORK

In this paper we presented the approach we follow to develop a social and semantic aware Web crawler for creating Web archives as community memories that revolve around events and the entities related to them. The need to make decisions during the crawl process with only a limited amount of information raises a number of issues. The division into different processing phases allows us to separate the initial complex extraction of events and entities from their faster but shallower detection at crawl time. Furthermore, it allows in the offline phase to learn more about particular events and topics the archivist is interested in and to get more insights about trustful content on the Social Web.

The implementation of the presented architecture is underway. Parts of the system are built upon existing technologies while other like the Social Web analysis need to be developed from scratch. Also, a number of detailed research questions need to be addressed. For example the typically limited set of reference pages and the limited time to detect topics, entities and events during crawling are open issues to be addressed. Also how the different extracted information, interaction patterns, etc., can be combined for prioritizing URLs is currently an open question. Moreover, the whole approach needs to be evaluated in real world scenarios; our current evaluation scenarios are related to the financial crisis and to the Rock am Ring concert series.

8. REFERENCES

[1] Apache Foundation. The Apache HBase Project. `http://hbase.apache.org/`, 2012.

[2] A. Arvidson and F. Lettenström. The Kulturarw Project - The Swedish Royal Web Archive. *Electronic library*, 16(2), 1998.

[3] R. Baeza-Yates, C. Castillo, M. Marin, and A. Rodriguez. Crawling a country: better strategies than breadth-first for web page ordering. In *Special interest tracks and posters of the 14th international conference on World Wide Web*, WWW '05, pages 864–872, New York, 2005. ACM.

[4] Blue Ribbon Task Force on Sustainable Digital Preservation and Access. Sustainable economics for a digital planet, ensuring long-term access to digital information, 2010.

[5] S. Chakrabarti, M. V. D. Berg, and B. Dom. Focused crawling: a new approach to topic-specific web resource discovery. In *Computer Networks*, pages 1623–1640, 1999.

[6] F. Chang et al. Bigtable: A distributed storage system for structured data. *ACM Transactions on Computer Systems (TOCS)*, 26(2):4, 2008.

[7] J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through url ordering. In *Proceedings of the seventh international conference on World Wide Web 7*, WWW7, pages 161–172, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V.

[8] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proc. of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, 2002.

[9] J. Dean and S. Ghemawat. MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

[10] D. Gomes, J. Miranda, and M. Costa. A survey on web archiving initiatives. In *In Proc. of the 15th Int. Conference on Theory and Practice of Digital Libraries*, TPDL'11, pages 408–420, Berlin, Heidelberg, 2011. Springer-Verlag.

[11] Internet Archive. NutchWAX. `http://archive-access.sourceforge.net/projects/nutch/`, 2012. [Online; accessed 2012-01-30].

[12] Internet Archive. Wayback. `http://archive-access.sourceforge.net/projects/wayback/` 2012. [Online; accessed 2012-01-30].

[13] J. Masanès. *Web archiving*. Springer, 2006.

[14] F. Menczer, G. Pant, and P. Srinivasan. Topical web crawlers: Evaluating adaptive algorithms. *ACM Trans. Internet Technol.*, 4:378–419, Nov. 2004.

[15] G. Mohr, M. Kimpton, M. Stack, and I. Ranitovic. Introduction to Heritrix, an archival quality web crawler. In *4th International Web Archiving Workshop (IWAW04)*, 2004.

[16] A. Ntoulas, J. Cho, and C. Olston. What's new on the web? The evolution of the web from a search engine perspective. In *In Proc. of the 13th Int. Conference on World Wide Web*, New York, USA, 2004.

[17] K. Shvachko, H. Kuang, S. Radia, and R. Chansler. The Hadoop distributed file system. In *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*, 2010.

[18] N. Tahmasebi, K. Niklas, T. Theuerkauf, and T. Risse. Using Word Sense Discrimination on Historic Document Collections. In *In Proc. of 10th ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, Surfers Paradise, Gold Coast, Australia, 2010.

[19] N. Tahmasebi, S. Ramesh, and T. Risse. First results on detecting term evolutions. In *In Proc. of 9th International Web Archiving Workshop in conjunction with ECDL 2009*, Corfu, Greece, 2009.

[20] C. Weiss, P. Karras, and A. Bernstein. Hexastore: sextuple indexing for semantic web data management. *Proceedings of the VLDB Endowment*, 1(1):1008–1019, 2008.

[14] `http://nutch.apache.org/`

[15] `http://livingknowledge-project.eu/`