Future Internet **2014**, *6*, 688-716; doi:10.3390/fi6040688

OPEN ACCESS

future internet

ISSN 1999-5903

www.mdpi.com/journal/futureinternet

Article

The ARCOMEM Architecture for Social- and Semantic-Driven Web Archiving

Thomas Risse [1],*, Elena Demidova [1], Stefan Dietze [1], Wim Peters [2], Nikolaos Papailiou [3], Katerina Doka [3], Yannis Stavrakas [3], Vassilis Plachouras [3], Pierre Senellart [4], Florent Carpentier [5], Amin Mantrach [6], Bogdan Cautis [4], Patrick Siehndel [1] and Dimitris Spiliotopoulos [7]

[1] L3S Research Center, Leibniz Universität Hannover, Hannover 30167, Germany;
E-Mails: demidova@L3S.de (E.D.); dietze@L3S.de (S.D.); siehndel@L3S.de (P.S.)

[2] NLP Group, Department of Computer Science, University of Sheffield, S1 4DP Sheffield, UK;
E-Mail: w.peters@dcs.shef.ac.uk

[3] ATHENA - Research and Innovation Center in Information, Communication and Knowledge Technologies, 15125 Maroussi, Athens, Greece; E-Mails: npapa@cslab.ntua.gr (N.P.); katerina@cslab.ece.ntua.gr (K.D.); yannis@imis.athena-innovation.gr (Y.S.); vassilis.plachouras@gmail.com (V.P.)

[4] CNRS LTCIT, Institut Mines-Télécom, Télécom ParisTech, 75634 Paris Cedex 13, France;
E-Mails: pierre@senellart.com (P.Se.); bogdan.cautis@u-psud.fr (B.C.)

[5] Internet Memory Foundation, 45 ter rue de la Révolution, 93100 Montreuil, France;
E-Mail: florent.carpentier@internetmemory.net

[6] Yahoo Research, 08018 Barcelona, Spain; E-Mail: amantrac@yahoo-inc.com

[7] Athens Technology Center (ATC), 15233 Halandri Athens, Greece; E-Mail: d.spiliotopoulos@atc.gr

* Author to whom correspondence should be addressed; E-Mail: risse@L3S.de;
Tel.: +49-511-762-17764; Fax: +49-511-762-17779.

External Editor: Andrew Hudson-Smith

Received: 15 April 2014; in revised form: 25 September 2014 / Accepted: 8 October 2014 / Published: 4 November 2014

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 (From

Collect-All Archives to Community Memories) architecture and implementation 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.

Keywords: 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 for preserving knowledge. The report *Sustainable Economics for a Digital Planet* [1] 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, the 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 [2] 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.* [3] 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 contrasted with the fact that, according to Gomes *et al.* [4], only approximately 40 Web archiving initiatives are active, which involve only about 270 people worldwide. Hence, the 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 dealing with novel requirements. Instead of following a "collect-all" strategy, archival organizations are trying to build community memories that reflect the diversity of information in which people are interested. 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 that 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 to 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.

In this paper, we give an overview of the crawler architecture implemented within the ARCOMEM project that addresses the requirements mentioned above. The paper goes beyond our initial work in [5] by providing more details about the different phases, the data management and analysis modules. It furthermore describes the implementation choices that lead to the final system. The ARCOMEM system has been published as open source [6].

The remainder 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. The implementation of the ARCOMEM system is described in Section 6. We discuss the state of the art in Web archiving and related fields in Section 7. Finally, Section 8 gives conclusions and an outlook for future work.

2. Use Cases and Requirements

In order to develop a Web crawler that 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 parliaments) 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 to rely only on trustworthy 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 broadcasters' own events 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: archivists and 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 in 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 an election, like the U.S. elections in 2012 or the German election in 2013. Journalists covering elections would like to have access to relevant content from official sources, journalistic sources, 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 elections, parties or the people involved?
- How are opinions distributed in relation to demographic user data and is it changing over time?
- Who are the most active Twitter users?
- 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, as well as journalists, political analysts and researchers, information and documentation regarding parliamentary issues. Besides traditional publications, the Web and the Social Web play an increasingly important role as an information source, since they provide 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 of the information provided in social media.

Through ARCOMEM, the Greek and Austrian parliaments aspire to transform their flat digital content archives into 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

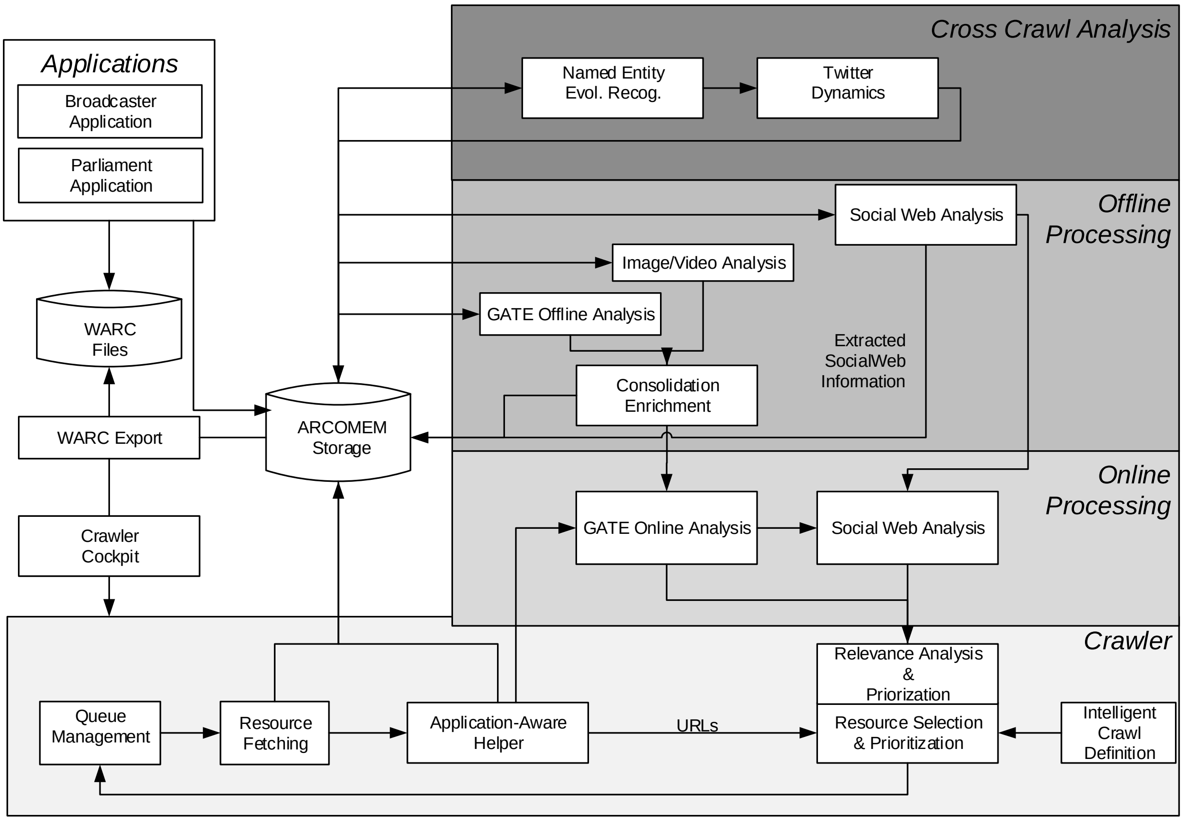
The requirements for the ARCOMEM system have been compiled in close collaboration with the broadcaster and parliament users and were based on the analysis of a number of use cases similar to those outlined above. The analysis phase identified a list of possible content sources for each use case, belonging 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 and 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. 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 cross crawl analysis, which revolve around the ARCOMEM storage, as depicted in Figure 1. The ARCOMEM storage, 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:

Figure 1. Overall architecture.



3.1. Crawling Level

At this level, the system decides on 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 storage 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 (ETOE) analysis and the analysis of the social contents operate over the cleansed data from the crawl, which are stored in the ARCOMEM storage. These processing tools perform linguistic, machine learning and natural language processing (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 storage and are available for further processing, text and data mining. After all of the relevant processing has taken place, the Web pages to be archived and preserved are selected in a semi-automatic way. For the selection, the user defines a lower threshold of the relevance score for the documents to be preserved. Finally, the selected original pages are transferred to the Web archive in the form of Web Archive (WARC) files (see Section 3.5). WARC [7] is an ISO standardized format for storing and archiving Internet content that aggregates crawled content into a single file. Besides the main content, WARC files also accommodate various meta-information (e.g., crawler information, crawl date and time, MIME type).

3.4. Cross Crawl Analysis Level

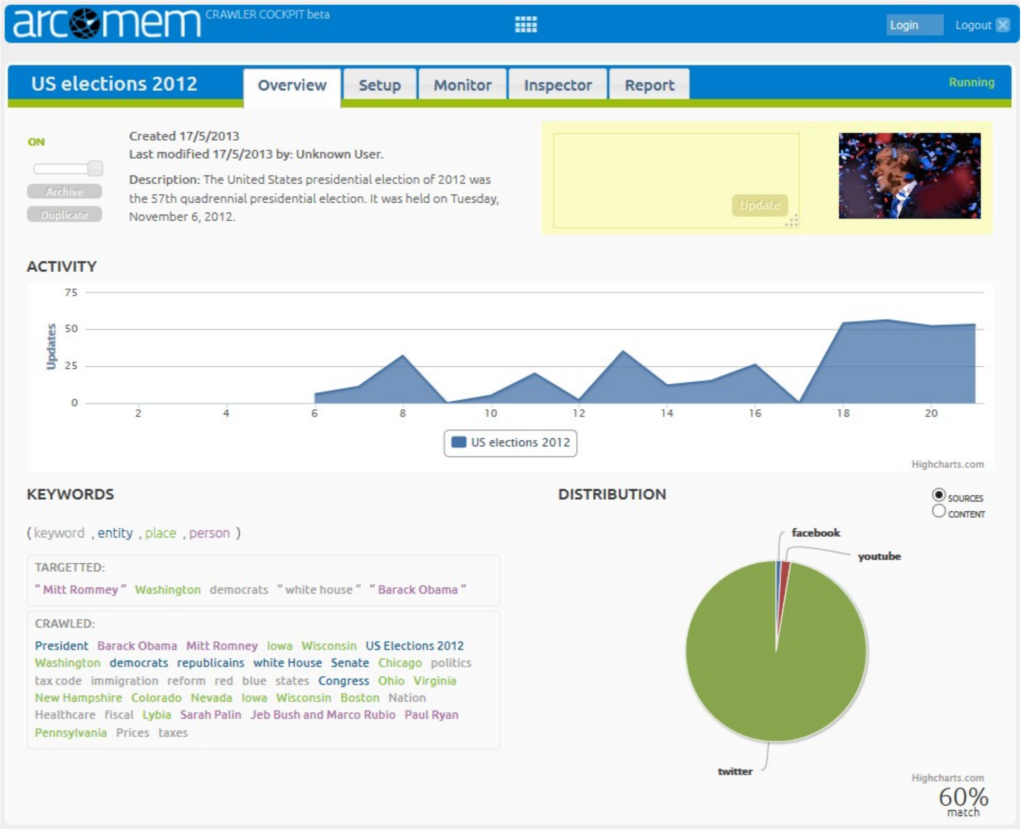
Finally, a more advanced processing step is taking place. This phase operates on collections of Web pages and Web objects (*i.e.*, components of a Web page, like images or videos) that have been collected over time in order to register the evolution of various aspects identified by the ETOE and Web analysis components. As such, it produces aggregate results that pertain to a group archive of objects, rather than to particular instances. Besides evolutions, every other complex analysis that requires combining several crawls can be operated within this phase.

3.5. Applications

We implemented customized methods to interact with the ARCOMEM crawler and ARCOMEM storage, for example to satisfy the use cases of Section 2. The Crawler Cockpit, as shown in Figure 2, allows archivist to specify or modify crawl specifications, monitor the crawl and do the quality assurance. The Crawler Cockpit 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 storage 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 Search and Retrieval Application (SARA) (see Figure 3) allows 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. More details about the SARA application are presented in [8].

Figure 2. Crawler Cockpit: crawl overview page.



Created 17/5/2013
Last modified 17/5/2013 by: Unknown User.

Description: The United States presidential election of 2012 was the 57th quadrennial presidential election. It was held on Tuesday, November 6, 2012.

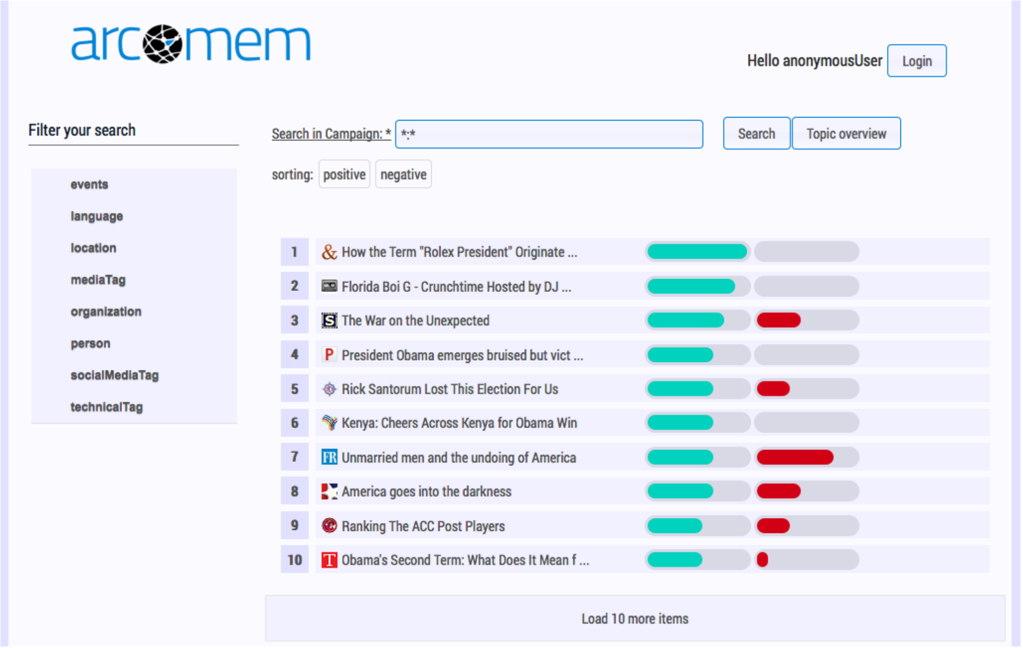
KEYWORDS

(keyword , entity , place , person)

TARGETTED:
" Mitt Rommey " Washington democrats " white house " " Barack Obama "

CRAWLED:
President Barack Obama Mitt Romney Iowa Wisconsin US Elections 2012 Washington democrats republicains white House Senate Chicago politics tax code immigration reform red blue states Congress Ohio Virginia New Hampshire Colorado Nevada Iowa Wisconsin Boston Nation Healthcare fiscal Lybia Sarah Palin Jeb Bush and Marco Rubio Paul Ryan Pennsylvania Prices taxes

Figure 3. Search and Retrieval Application (SARA): result page.



Filter your search

- events
- language
- location
- mediaTag
- organization
- person
- socialMediaTag
- technicalTag

Search in Campaign: * *:*

sorting: positive negative

1 How the Term "Rolex President" Originate ...
2 Florida Boi G - Crunchtime Hosted by DJ ...
3 The War on the Unexpected
4 President Obama emerges bruised but vict ...
5 Rick Santorum Lost This Election For Us
6 Kenya: Cheers Across Kenya for Obama Win
7 Unmarried men and the undoing of America
8 America goes into the darkness
9 Ranking The ACC Post Players
10 Obama's Second Term: What Does It Mean f ...

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 in OWL (Web Ontology Language) [9] that reflects the informational needs of ARCOMEM. Then, from the perspective of data interoperability, access and reuse we have concentrated on embedding the ARCOMEM conceptual vocabulary in the wider semantic Web context. We have done this in several ways.

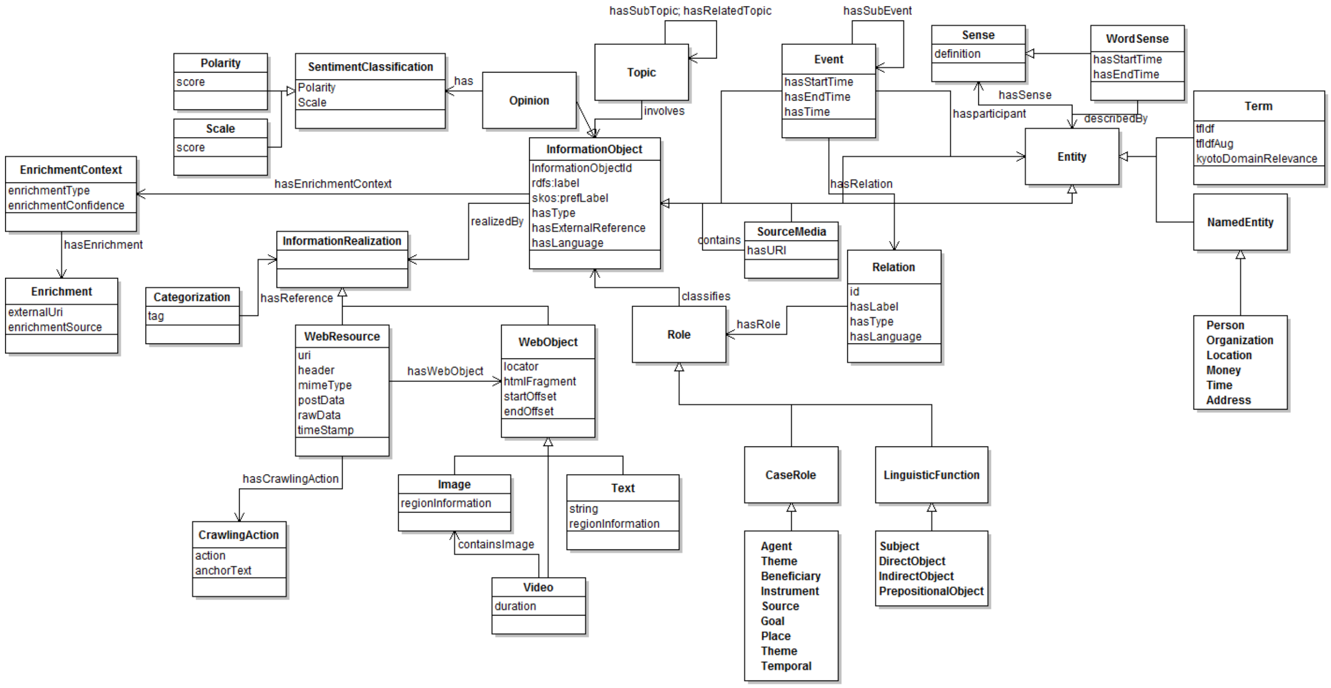
First, we have ensured interoperability with the Open Archival Information System [10] (OAIS, ISO 14721). OAIS constitutes a conceptual framework for long term digital information preservation and access. As such, it provides a central point of entry into OAIS related standardization, and a basis for comparing the data models of digital information preserved by archives. Therefore, interoperability with the OAIS information model is a prerequisite for the ARCOMEM data to be taken into account for the purpose of long term preservation.

In addition, the ARCOMEM information concepts have been linked to concepts from the wider semantic Web context including the LOD (Linked Open Data) cloud, in order to harmonize the semantic coverage and ensure a proper embedding of the ARCOMEM conceptual vocabulary in the wider semantic Web context. The basic idea behind Linked Data is the use of globally valid identifiers for referencing resources. This reduces the integration effort required for semantic interoperability. Linked Data are typically published in W3C-Standards RDF (Resource Description Framework) [11] or OWL [9]. ARCOMEM ontology elements have been linked with elements from foundational ontologies such as Dolce UltraLight [12] and officially established or de facto standard vocabularies for conceptual constructs such as event, e.g., Linked Open Descriptions of Events (LODE) [13], Event-Model-F [12].

The central concepts in ARCOMEM's 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.

Figure 4 illustrates the resulting data model in the form of a simplified UML notation in the form of classes and their relations. Open arrows represent subClassOf relations, whereas closed arrows indicate any other type of relation. The ARCOMEM data model is represented in OWL. This enables structured access to its information content through SPARQL queries. The query structure relies on statements expressed in the form of subject-predicate-object *triples*, denoted as $\{S, P, O\}$ [14].

Figure 4. The ARCOMEM Data Model.



4.2. ARCOMEM Storage

The ARCOMEM storage 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: (1) binary data, in the form of Web objects, which represent the original content collected by the crawler; and (2) 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 storage 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.

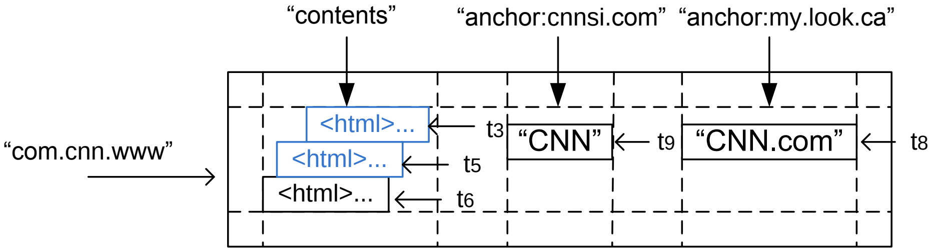
To cover the functional requirements, we have designed and implemented a storage module consisting of two components: the object store and the knowledge base. Both of them rely on distributed solutions that combine the MapReduce [15] programming model and NoSQL databases to cover the non-functional requirements. 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 a low cost.

4.2.1. The ARCOMEM Object Store

The object store is responsible for storing and serving the raw content harvested by the crawler. It relies on Apache HBase [16], a NoSQL database written in Java and modeled after Google's BigTable [17]. 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 data, HBase keeps its data in a distributed filesystem called Hadoop Distributed File System (HDFS) [18]. 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 5. 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 the *a posteriori* addition of attributes, as well, allowing the data model to be enriched on demand.

Figure 5. Sample content of the object store.



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.

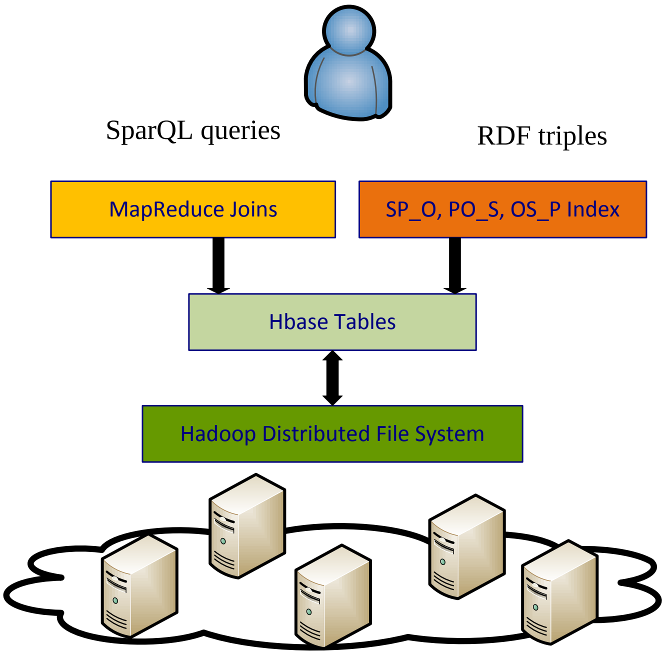
4.2.2. The ARCOMEM Knowledge Base

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 Section 4.1) that interlinks ETOEs and points 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, H_2RDF [19,20], 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 and MapReduce.

More specifically, H_2RDF implements an enhanced version of the centralized Hexastore indexing scheme [21] over HBase. Hexastore is an RDF store that creates six 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 data for any triple pattern with minimal cost. Taking this one step further, our distributed knowledge base creates three indices ultimately stored in HBase tables. 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 either as MapReduce jobs or as centralized jobs according to their cost. This ensures interactive response times for small selective queries, as well as scalability and distributed execution for large, non-selective queries. Figure 6 pictorially presents the knowledge base architecture. Extensive experiments presented in [19,20] prove the ability of our knowledge base, H_2RDF, to scale to billions of triples and to handle concurrent user requests, achieving fast response times.

Figure 6. The knowledge base architecture.



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 content with useful semantic information. We discuss content analysis, analysis of the Social Web, dynamics analysis, data enrichment and crawler guidance.

5.1. Content Analysis

In order for archivists to judge the content of a particular Web document and to decide whether it should be archived, they need to be able to assess this content. The aim of the content analysis module is the extraction and detection of informational elements, called ETOEs (entities, topics, opinions and events) from Web pages (see 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. In the online phase, a newly crawled Web page will be analyzed to detect the terms of the search string. Afterwards, the coverage of the page will be compared with the crawl specification. This information is used by the intelligent crawler to prioritize the extracted URLs. A high overlap between the terms on the newly crawled page and the terms describing the search string in the crawl specification indicate a highly relevant page and, thus, a high priority for the page and its URLs.

In the offline phase, the extraction of ETOEs from Web pages is performed by robust and adaptable processing tools developed within the GATE (General Architecture for Text Engineering) architecture [22]. GATE is a framework for language engineering applications, which supports efficient and robust text processing. GATE uses natural language processing (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, morphological analysis and verb and noun phrase chunking). This is then followed by the extraction of both named entities using ANNIE (a Nearly-New Information Extraction System) [22] and term candidates using TermRaider [23]. 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 the 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.

5.2. Data Enrichment and Consolidation

As the content analysis extracts structured data from unstructured resources, such as text and images, the generated data is highly 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 knowledge base as separate entities described according to the ARCOMEM data model.

Data enrichment and consolidation in ARCOMEM follow two aims: (1) to enrich existing entities with related publicly-available knowledge; and (2) to identify data correlations within the Web archives created by ARCOMEM by aligning extracted entities with reference datasets. Both (1) and (2) exploit publicly-available data from the Linked Open Data (LOD) cloud, which offers a vast amount of data of both a 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 [24] and Freebase [25] as reference datasets, though this approach does not depend on any particular dataset and is applicable to additional and more domain-specific datasets, e.g., event-specific ones. DBpedia and Freebase are particularly well-suited for the enrichment of Web crawls, 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. For example, Freebase currently contains more than 22 million entities in more than 100 domains and more than 350 million facts about these entities.

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 (“Ireland”, “Republic of Ireland”, “Irland”), all referencing 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, the Web objects that these entities originate from) can be clustered directly by identifying joint enrichments of 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 one 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 to 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 (enriched with a reference to http://dbpedia.org/resource/Dublin, which directly links to http://dbpedia.org/resource/Ireland), James Joyce (http://dbpedia.org/resource/James_Joyce) or the IRA (http://dbpedia.org/resource/Irish_Republican_Army), can be associated and correlated simply by analysing the DBpedia graph to identify correlations between existing enrichments.

While in a large graph, such as DBpedia or Freebase, any node 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. To this extent, we developed approaches that analyse the strength of entity relationships in the data graphs and uncover important entity relationships [26]. These approaches can be used to cluster closely-related entities and the Web objects that they originate from in ARCOMEM Web archives.

Furthermore, enrichments can provide an additional hierarchical view on the extracted entities and, thus, archived Web objects. Both DBpedia and Freebase possess rich category structures that can be interconnected with the archived Web objects using reference entities. For example, in DBpedia "Ireland" is classified as a region in Northern and Western Europe. This classification can help in finding other entities located in these regions within the Web archive, as well as the archived Web objects related to these entities. In addition to hierarchical structures natively provided by the LOD datasets, further hierarchies can be obtained by mapping the schemas of the reference datasets used by the enrichment process to external ontologies. A good illustration of this approach is YAGO+F [27]—a mapping we created to interconnect Freebase schema with the rich category structure of the YAGO ontology [28]. The YAGO hierarchy includes more than 20 levels and can enable views on the archived content at many different levels of abstraction.

5.3. Social Web Analysis

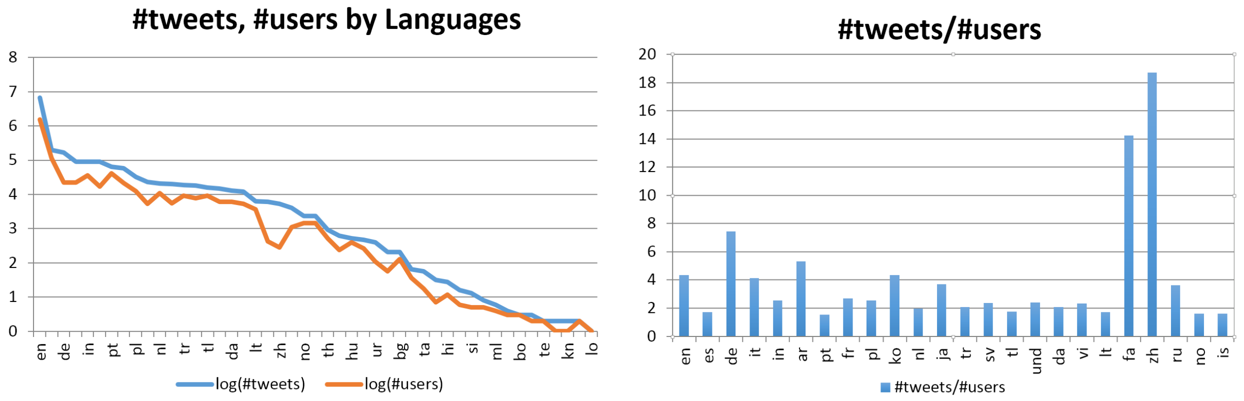
An important aspect is that content is created in a specific social context, and thus, it reflects particular socio-historical moments. With the advent and widespread use of the Web, it is clear that societies around the world are increasingly connected: events in one part of the world are known at very distant locations with little latency, and once news breaks out, everybody has contextual information at some level. The meaning of this information can vary significantly, but one thing that characterizes these current flows of information is that increasingly, people are empowered to create, modify and share content. In many ways, these actions create the context around events. One of the big challenges for archiving and for journalism is therefore being able to leverage this contextual information. On the one hand, context defined as the circumstances that form the setting for an event, statement or idea and in terms of which it can be fully understood and assessed is often described by information in social media (e.g., reactions, connections between individuals that talk about the event, *etc.*). On the other hand, such context can serve to determine the importance of particular units of information and to explain them in future settings. In the following, we briefly describe techniques and approaches concerned with Social Web archive contextualization. In particular, we describe our work on cultural dynamics, which refers to gaining insights into cultural patterns in social media, the detection of experts on Twitter, the reaction of people in social media to news articles and context-aware social search.

5.3.1. Cultural Dynamics

The purpose of this work is to investigate techniques and large-scale analysis methods on social media that can be helpful in Social Web archive contextualization. In particular, this means that the results of analysis, such as those presented here, could be an integral component of the archive and that the results of such analysis could also help guide the archiving process, as well as the design of algorithms for the other use cases. As we note below, there are significant differences in the general characteristics of the Twitter network in different countries. Such differences should be taken into account in designing crawling and preservation strategies and can help explain differences in how information spreads in such networks. In the journalism case, for instance, insights into those networks could help explain and determine reliable sources of information [29].

For illustration, we present the results of the analysis of the cultural dimensions of a set of tweets crawled by the ARCOMEM system for the period November 1–11, 2012, for a set of target keywords related to the U.S. Elections 2012. Figure 7(a) shows the distributions of the languages for tweets and for users on the U.S. elections crawl. Without surprise, the distributions exhibit a power-law indicating as the dominant language mainly English. Interestingly, the user engagement (Figure 7(b)) shows that Farsi speakers (Iranian) and Mandarin speakers (Chinese) are the most engaged people while tweeting on the U.S. elections.

Figure 7. (**a**) Distribution of the number of tweets for each language. Power law distribution of the number of tweets, and of users by languages. (**b**) Number of tweets by user for each language. Measure of the user engagement.



(**a**) (**b**)

5.3.2. Twitter Domain Expert Detection

The information provided by social networks like Twitter covers nearly all areas of interest. Besides the fact that useful information is provided, it is still a hard task to discover if the source is trustworthy and if the expertise of the tweeting user is high regarding the topic. The goal of the Twitter domain expert module is to analyze and measure the expertise of users in Twitter. This information can be used to give higher priorities to the crawling module for the Web sites suggested by experts in a certain domain. The underlying assumption is that users who are experts in a certain domain link Web sites that are more useful to describe a certain topic or event. We define a domain expert as a person who has deeper knowledge regarding a certain domain than the average user. In particular, we analyze if the tweets published by a certain user contain keywords that are very specific and part of a professional vocabulary. The approach is based on the methods presented in [30].

The biggest challenge in this task is to find appropriate algorithms and metrics for defining whether a word is part of this professional vocabulary or not. Besides that, we also need to know in which area the user is an expert. The method we developed to solve this task is based on the vast amount of information provided by Wikipedia. We use the link and category information supplied by Wikipedia to define the topic and the expertise level inherent in certain terms. A more detailed description of the categorization process is given in [31]. For defining the expertise level of a given concept, we use the distance to the root within the Wikipedia category graph. We use the Wikipedia Miner Toolkit [32] to link tweets with Wikipedia articles to derive afterwards the most probable categorization of the tweet. The annotation of the Tweets is done using the methods [33] provided by the Wikipedia Miner Toolkit on a single tweet level. In the final aggregation step, all tweets of users are aggregated to generate the final user profile. This profile can be based on only a single tweet or on several tweets, depending on the availability of tweets for the user. When using the Twitter Streaming API, in most cases, the tweets are all related to a set of predefined keywords, and due to that, only a limited number of tweets per user is available. The generated user profiles display the topics a user talks about, as well as the expertise in a certain topic.

5.3.3. Responses to News

Social media responses to news have increasingly gained importance, as they can enhance a consumer's news reading experience, promote information sharing and aid journalists in assessing their readership's response to a story. Responses to real-world events often come in the form of short messages, referring to specific topics, content or information sources. Such messages serve several purposes: to share a particular news article, to express an opinion about the ongoing events or to add or refute information about the topic or the mentioned article. The extensive use of social media during recent major events (e.g., the Arab Spring and the financial crisis) shows that its use in these situations has become pervasive. On Twitter, for instance, a significant share of all tweets posted concerns news events [34]. Considering this large volume of messages being posted in the context of news, keeping track of messages that refer to the most popular articles can easily become overwhelming. This information overload motivates the development of automatic systems that select and display only the most interesting messages.

The selection system takes as input a news article and all of the social media responses referring to that article and outputs the most interesting subset of responses. Our work is motivated by a variety of existing and future applications in which interesting social media responses could be automatically coupled with traditional news content, e.g., for displaying social responses near a news article for an enhanced reading experience. Even though quantifying the interestingness of a selection of messages is inherently subjective, we postulate that an interesting response set consists of a diverse set of informative, opinionated and popular messages written to a large extent by authoritative users. By decomposing the notion of interestingness into these indicators, we can pose the task of finding an interesting selection of messages as an optimization problem. We aim at maximizing an objective function that explicitly models the relationship among the indicators, thus producing selections that the typical person finds most interesting.

Our method presented in [35] considers multiple content, social and user features to infer the intrinsic level of informativeness, opinionatedness, popularity and authority of each message, while simultaneously ensuring the inclusion of diverse messages in the final set. We evaluate our approach through both human and automatic experiments and demonstrate that it outperforms the state of the art. In addition, we perform an in-depth analysis of the human evaluations, shedding light on the subjectivity and perception of interestingness in this particular task.

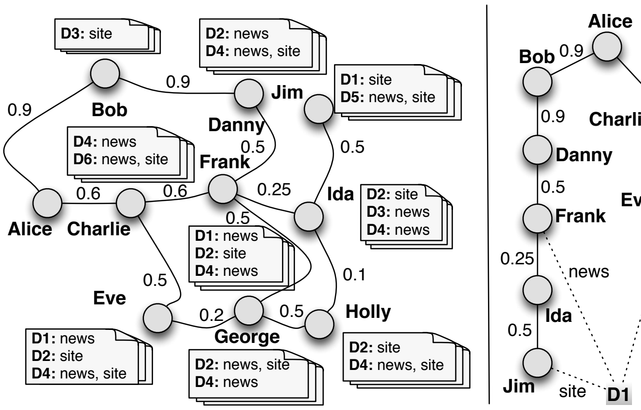
5.3.4. Context-Aware Search

We consider the problem of top-k retrieval in social tagging (or bookmarking) systems, with a focus on efficiency, targeting techniques that have the potential to scale to current applications on the Web in an online context where the network, the tagging data and even the seekers' search ingredients can change at any moment. In this context, a key sub-problem for top-k retrieval that we need to address is computing scores of top-k candidates by iterating not only through the most relevant items with respect to the query, but also (or mostly) by looking at the closest users and their tagged items.

We associate with the notion of social network a rather general interpretation, as a user graph whose edges are labeled by social scores, which give a measure of the proximity or similarity between two users. These are then exploitable in searches, as they say how much weight one's tagging actions should have in the result build-up. For example, even for applications where an explicit social network does not exist or is not exploitable, one may use the tagging history to build a network based on similarity in tagging and items of interest. While we focus mainly on social bookmarking applications, we believe that these represent a good abstraction for other types of social applications, to which our techniques could apply.

As an example, let us consider the collaborative tagging configuration of Figure 8 (left). Users have associated lists of tagged documents, and they are interconnected by social links. Each link is labeled by its (social) score, assumed to be in the $(0, 1]$ interval. Let us consider user Alice in the role of the seeker. The user graph is not a complete one, as the figure shows, and only two users have an explicit social score with respect to Alice. For the remaining ones, Danny, …, Jim, only an implicit social score could be computed from the existing links if a precise measure of their relevance with respect to Alice's queries is necessary in the top-k retrieval.

Figure 8. Collaborative tagging scenario and its social network (**left**); some paths from a seeker towards a relevant item (**right**).



Let us assume that Alice looks for the top two documents that are tagged with $news$ and $site$. Looking at Alice's immediate neighbors and their respective documents, intuitively, $D3$ should have a higher score than $D4$, since the former is tagged by a more relevant user (Bob , having the maximal social score relative to Alice). If we expand the search to the entire graph, e.g., via a shortest paths-like interpretation, the score of $D4$ may however benefit from the fact that other users, such as Eve or even $Holly$, also tagged it with $news$ or $site$. Furthermore, documents, such as $D2$ and $D1$, may also be relevant for the top two result, even though they were tagged only by users who are indirectly linked to Alice.

Figure 8 (right) gives a different perspective on our scenario, illustrating how Alice can reach one of the relevant documents, $D1$, by following three paths in the social network. Intuitively, an aggregation of such paths from a seeker towards data items will be used in the scoring model. Under certain model assumptions that are omitted here, pertaining to social and textual relevance, the top two documents for Alice's query will be, in descending score order, $D4$ and $D2$. We provide the underlying model and algorithms for building this type of answer.

In [36], we presented our TOPKS algorithm for top-k answering in collaborative tagging, which has the potential to scale to current online applications, where network changes and tagging are frequent. The contribution of our approach is three-fold:

- we allow full scoring personalization, where potentially each user of the system can define their own way to rank items.
- we can iterate over the relevant users more efficiently, sparing the potentially huge disk volumes required by existing techniques, while also having the potential to run faster; as a bonus, most social networks could fit in main memory in practice.
- social link updates are no longer an issue; in particular, when the social network depends on user actions (e.g., the tagging history), we can keep it up-to-date and, by it, all the proximity values at any given moment, with little overhead.

Based on this, our top-k algorithm is sound and complete, and when the search relies exclusively on the social weight of tagging actions, it is instance optimal, *i.e.*, it visits a minimal number of users, in a large and important class of algorithms. Extensive experiments on real-world data from Delicious and Twitter show that TOPKS performs significantly better than state-of-the-art techniques, up to two-times faster.

Going further, since in real-world online applications, the joint exploration of the social and textual space may remain costly, even by optimal algorithms, we consider directions for approximate results. More specifically, these are motivated by the fact that estimates for termination conditions may be too loose in practice and that exact shortest paths, like computations, may be too expensive in a large network. Our approaches present the advantages of negligible memory consumption, relying on concise statistics and pre-computed information about the social network and proximity values (via landmarks), and reduced computation overhead. Moreover, these statistics can be maintained up to date with little effort, even when the social network is built based on tagging history. Experiments show that approximate techniques can drastically improve the response time of the exact approach, even by an order of magnitude, with reduced impact on precision. More details about the approach can be found in [36,37].

5.4. Cross Crawl Analysis

The aim of cross crawl analysis is to enable the analysis across a number of crawls. Within ARCOMEM, this has been used to enable temporal analytics of the crawled content. However, any other kind of analysis is possible in this phase. In the following subsections, we will briefly present modules for detecting the evolution of entities, as well as the dynamics on Twitter.

5.4.1. Language Dynamics

High impact events, political changes and new technologies are reflected in our language and lead to the constant evolution of terms, expressions and names. There are several kinds of language evolution, among others spelling variations, name changes and concept changes. We focus on named entity evolution, the detection of name changes, as it has a high impact, for example in information retrieval in archives, as well as linking of entities over long time spans. In [38], we presented NEER (Named Entity Evolution Recognition) —an unsupervised method for named entity evolution recognition. While other approaches rely on external resources, like DBPedia or Freebase, NEER is able to detect evolution independent of such knowledge resources. This allows the detection of evolution even if it has not been described in any knowledge resource. This is especially important for Social Web content, since language in social media is very dynamic and the documentation of language changes is not guaranteed.

Named entity changes are typically associated with significant events concerning the entities, which lead to increased attention. We use this property in NEER to pinpoint change periods and to detect those using burst detection. After identifying change periods for an entity, we create a context for each period by extracting all documents that mention the entity or any part of it and are published in the year corresponding to the change period. We extract nouns, noun phrases and named entities. Noun phrases are used to capture more information and create richer contexts around entities. Based on the contexts, direct co-references are identified and similar co-references are merged into co-reference classes. Ultimately, the graphs are consolidated by means of the co-reference classes. Afterwards, filtering methods filter out false co-references that do not refer to the query term. For this purpose, statistical, as well as machine learning (ML)-based filters were introduced. A comparison of the methods revealed their strengths and weaknesses in increasing precision while keeping a high recall. The ML approach performed best with noticeable precision and recall of more than 90%. While it is possible to deliver a high accuracy with NEER + ML, training the needed ML classifier requires manual labelling. More details on term evolution can be found in [38,39].

5.4.2. Twitter Dynamics

The aim of the Twitter Dynamics module is to provide a flexible way for exploring and selecting tweets based on the strength of the associations between interesting terms and the evolution of those associations across time. In many cases, it is important to select the correct set of tweets to feed into subsequent analysis processes. This is usually performed through search queries to the API of Twitter for specific terms that relate to a domain of interest. The selection of terms, however, can have an important impact on the quality of the results. O’Connor *et al.* [40] observe such an impact when they use variations of words to correlate a sentiment score from tweets with an index of consumer confidence. The same issue is also mentioned in the context of sentiment analysis by Jiang *et al.*, who suggest that classifying the sentiment expressed about a target keyword may be enhanced by also classifying the sentiment expressed for other related keywords [41]. Consequently, we argue that it is important to be able to explore the context of terms, to create datasets and to perform complex analysis tasks based on term associations. Currently, there is a gap between this need and the functionality that online social network APIs offer.

In the Twitter dynamics module, we designed and implemented a model and a number of query operators [42] that enable users to explore the context of keywords in data obtained from online social networks and microblogging services. The model represents terms as nodes and associations between terms as directed edges annotated with the association weight and the corresponding time span. We introduced the following query operators: filter for asserting selection conditions, fold for changing the time granularity, merge for grouping across time and join. By combining query operators, users can discover the relationships between keywords (as they are implied by their occurrence in tweets) and their evolution in time. The operators also allow one to express complex conditions on the associations of terms that take temporal aspects into account and to retrieve the subset of tweets that satisfy these conditions. For example, a journalist, who is exploring Twitter data consisting of timestamped hashtags, can issue the following query concerning the financial crisis:

"For the period during which there is a strong association between hashtags #crisis and #protest, which other hashtags are associated with both #crisis and #protest? Which are the relevant tweets?"

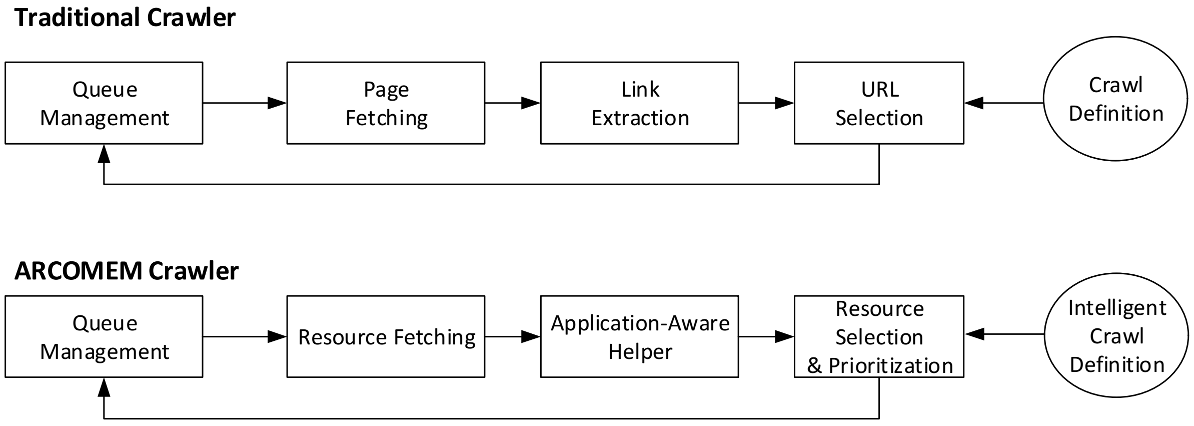
The above example query will first identify the time span during which there is a strong association between the hashtags #crisis and #protest. Next, other hashtags that are strongly related to both #crisis and #protest will be identified. Finally, the relevant tweets that contain any of the identified hashtags in the corresponding time span will be returned. We assume that two terms are associated when they co-occur in the same tweet. A detailed description of the model and the operators can be found at [42].

5.5. Crawler Guidance

Consider first the simplified view of the architecture of a traditional Web crawler depicted in Figure 9 (top). In a traditional Web crawler, such as Heritrix [43], 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 9. 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 simply accessible by a simple HTTP GET request, but only by other means: a succession of such requests, a POST request, the use of an API (which is a very common situation on Social Web networks) or the selection of 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 object store for use by analysis modules and archivist tools.

Figure 9. Traditional processing chain of a Web crawler compared to the ARCOMEM approach.



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 are, depending on their nature (embeds or other Web resources), 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 and prioritization module that makes use of the intelligent crawling definition and 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. More details about the crawling strategy can be found in [44].

6. Implementation

The ARCOMEM system has been implemented as described in the previous sections. To achieve storage scalability and benefit from distributed processing, we used the HBase and Hadoop combination. Both the online and offline analyses require running analysis modules in map-reduce jobs over the Web data stored in HBase and to write outputs in either HBase along the Web data or as triples in the

ARCOMEM knowledge base (see Section 4.2). We created the ARCOMEM framework to ease these tasks. Since Hadoop and HBase are written in Java and as it turned out to be an appropriate language for the analysis modules, we wrote the ARCOMEM framework in Java, too. The framework features a flexible configuration system that allows one to describe the dependencies between processes. Any analysis can be split into as many map-reduce and local processes as needed, and the framework will run them in an order compatible with the dependency graph. The dependencies between different analysis modules can be expressed in the same way.

When dealing with Web data, robustness is a prime concern, since a wide variety of content will be encountered. We tried to isolate the code dealing with Web content as much as we could: we run certain processes in a separate thread and limit the time and memory available to them. When a resource exceeds these limits, we mark it as problematic in HBase and skip it on subsequent runs. As a last resort, when a whole task fails repeatedly, it is skipped (we make it return immediately with a success status), and the rest of the job is allowed to proceed. We initially studied the use of a trigger on Put operations to run the online analysis as we imported Web content into HBase. Since it requires running analysis code directly in the region servers of Hadoop, we preferred to use a map-reduce job after import, trading extra latency for better isolation. This also helps in reducing the amount of code, by making the online analysis a special case of offline processing.

Performance is another major concern when dealing with Web-scale data. We used per-resource calculations implemented in the map phase everywhere we could. When a corpus was needed, we chose to process small batches in the reducer with a constant number of documents independent of their size. The triple upload to the ARCOMEM knowledge base is done in batches when applicable. In addition, we profiled the online analysis and optimized it drastically, including making some quality/time trade-offs.

7. Related Work

Since 1996, several projects have pursued Web archiving (e.g., [45]). The Heritrix crawler [43], jointly developed by several Scandinavian national libraries and the Internet Archive through the International Internet Preservation Consortium (IIPC) [46], is a mature and efficient tool for large-scale, archival-qualityc̃rawling.

The method of choice for memory institutions is client-side archiving based on crawling. This method is derived from a search engine crawl and has been evolved by the archiving community to achieve a greater completeness of capture and an increase in the 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 [47]. These issues were addressed in the European project, LiWA (Living Web Archives) [48].

The task of crawl prioritization and focusing is the step in the crawl processing chain that 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 in order to perform a crawl according to relevance. 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 [49,50], but since these generic approaches do not cover specific information needs, focused or topical crawls have been developed [51,52]. Recently, Laranjera *et al.* [53] published a comparative study about the quality of different crawling strategies for gathering comparable corpora on specific topics. According to this stud,y variants of the best-N-first strategy perform best depending on quality of the seed list. The best-N-first (BFS) strategy prioritizes links according to the relevance of their parent page and fetches the first n pages of the queue. In contrast, the ARCOMEM crawler uses entities instead of n-grams to determine the relevance of a page [44].

Beside domain or topical crawl strategies "user-centric" [54] harvesting strategies also exist, such as Adscape [55]. The aim of Adscape is to maximize the number of harvested unique advertisements on the Web. This is achieved by dynamically generating user profiles to collect personalized advertisements from Web pages.

Recently, the collection of Social Web content has gained more interest. Many services, such as Twitter, YouTube or Flickr, provide through their APIs access to structured information about users, user networks and created content. Data collection from these services is not supported by standard Web crawlers. Usually it is conducted in an *ad hoc* manner, although some structured approaches exist [56–58]. However, these platforms focus on API access and do not archive Web pages linked from Social Web platforms.

With the success of the semantic Web and linked data, also the harvesting of these resources has gained more interest. LDSpider [59] is a crawler to traverse and harvest the Linked Data Web. A similar approach is followed by the Slug [60].

A limited set of tools exist for accessing Web archives, like NutchWAX and Wayback. NutchWAX [61] is based on Apache Nutch [62]—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. The Portuguese Web Archive information retrieval system [63] adapted NutchWAX to support versions of the same URL across time and to improve the ranking of results.

A URL-based access to Web archives is provided by Wayback [64]. Wayback 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 allow accessing and browsing by a number of different facets.

With the availability of long-term Web archives, increasing interest in the temporal analysis of these collections can be observed. Within the LAWA (Longitudinal Analytics of Web Archive data) project [65], temporal Web analytics has been investigated with respect to content semantics and processing scalability. The BUDDHA (Big UK Domain Data for the Arts and Humanities) [66] project aims at developing a theoretical and methodological framework for the analysis of Web archives for arts and humanities researchers.

8. Conclusions and Future Work

In this paper, we presented the approach that we followed 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, in the offline phase, ARCOMEM allows the enrichment of the archive with additional social and semantic information that eases the use of content in applications.

The system has been implemented on Hadoop and HBase following the MapReduce paradigm to ensure good scalability. It is published as open source [6]. Evaluations of the different components and phases will be published in separate papers.

Acknowledgments

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

Author Contributions

Thomas Risse contributed to Sections 1, 3, 5.4 and 8. Pierre Senellart contributed to Section 1 and Section 5.5. Dimitris Spiliotopoulos contributed to Section 2. Yannis Stavrakas contributed to Section 2 and Section 5.4. Wim Peters contributed to Section 4 and Section 5. Nikolaos Papailiou and Katerina Doka contributed to Section 4.2. Elena Demidova and Stefan Dietze contributed to Section 5.2. Amin Mantrach, Patrick Siehndel, and Bogdan Cautis contributed to Section 5.3. Vassilis Plachouras contributed to Section 5.5. Florent Carpentier contributed to Section 6. All contributed to Section 7.

Conflicts of Interest

Thomas Risse and Wim Peters are co-editors of the Special Issue on Archiving Community Memories.

References

1. Blue Ribbon Task Force on Sustainable Digital Preservation and Access. Sustainable Economics for a Digital Planet, ensuring Long-Term Access to Digital Information. Available online: http://brtf.sdsc.edu/biblio/BRTF_Final_Report.pdf (accessed on 23 October 2014).
2. ARCOMEM: Archiving Communities Memories. Available online: http://www.arcomem.eu/ (accessed on 10 April 2014).
3. Ntoulas, A.; Cho, J.; Olston, C. What's New on the Web? The Evolution of the Web from a Search Engine Perspective. In Proceedings of the 13th International Conference on World Wide Web, New York, NY, USA, 17–20 May 2004.
4. Gomes, D.; Miranda, J.; Costa, M. A Survey on Web Archiving Initiatives. In Proceedings of the 15th International Conference on Theory and Practice of Digital Libraries (TPDL), Berlin, Germany, 26–28 September 2011.
5. Risse, T.; Dietze, S.; Peters, W.; Doka, K.; Stavrakas, Y.; Senellart, P. Exploiting the Social and Semantic Web for Guided Web Archiving. In Proceedings of the 16th International Conference on Theory and Practice of Digital Libraries (TPDL), Paphos, Cyprus, 23–27 September 2012.
6. The ARCOMEM Consortium. ARCOMEM system release. Available online: http://sourceforge.net/projects/arcomem/ (accessed on 25 July 2014).

7. ISO. ISO 28500:2009 Information and Documentation—WARC File Format. Available online: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44717 (accessed on 23 October 2014).
8. Demidova, E.; Barbieri, N.; Dietze, S.; Funk, A.; Holzmann, H.; Maynard, D.; Papailiou, N.; Peters, W.; Risse, T.; Spiliotopoulos, D. Analysing and Enriching Focused Semantic Web Archives for Parliament Applications. *Futur. Internet* **2014**, *6*, 433–456.
9. McGuinness, D.L.; van Harmelen, F. (eds.) OWL Web Ontology Language. Available online: http://www.w3.org/TR/owl-features/ (accessed on 23 October 2014).
10. Lee, C. Open Archival Information System (OAIS) Reference Model. Available online: http://www.tandfonline.com/doi/abs/10.1081/E-ELIS3-120044377 (accessed on 23 October 2014).
11. Resource Description Framework (RDF). Available online: http://www.w3.org/RDF/ (accessed on 23 October 2014).
12. A. Scherp, T. Franz, C.S.; Staab, S. F-A Model of Events based on the Foundational Ontology DOLCE + DnS Ultralight. In Proceedings of the International Conference on Knowledge Capturing (K-CAP), Redondo Beach, CA, USA, 1–4 September 2009.
13. Shaw, R.; Troncy, R.; Hardman, L. LODE: Linking Open Descriptions of Events. In Proceedings of the 4th Asian Semantic Web Conference (ASWC), Shanghai, China, 6–9 December 2009.
14. The ARCOMEM Consortium. ARCOMEM Data Model. Available online: http://www.gate.ac.uk/ns/ontologies/arcomem-data-model.owl (accessed on 23 October 2014).
15. Dean, J.; Ghemawat, S. MapReduce: Simplified data processing on large clusters. *Commun. ACM* **2008**, *51*, 107–113.
16. Apache Foundation. The Apache HBase Project. Available online: http://hbase.apache.org/ (accessed on 23 October 2014).
17. Chang, F.; Dean, J.; Ghemawat, S.; Hsieh, W.C.; Wallach, D.A.; Burrows, M.; Chandra, T.; Fikes, A.; Gruber, R.E. Bigtable: A Distributed Storage System for Structured Data. *ACM Trans. Comput. Syst.* **2008**, doi:10.1145/1365815.1365816.
18. Shvachko, K.; Kuang, H.; Radia, S.; Chansler, R. The Hadoop Distributed File System. In Proceedings of the IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), Lake Tahoe, NV, USA, 3–7 May 2010.
19. Papailiou, N.; Konstantinou, I.; Tsoumakos, D.; Koziris, N. H2RDF: Adaptive Query Processing on RDF Data in the Cloud. In Proceedings of the 21st International Conference Companion on World Wide Web (Companion Volume), Lyon, France, 16–20 April 2012.
20. Papailiou, N.; Konstantinou, I.; Tsoumakos, D.; Karras, P.; Koziris, N. H2RDF+: High-performance distributed joins over large-scale RDF graphs. In Proceedings of the IEEE International Conference on Big Data, Santa Clara, CA, USA, 6–9 October 2013.
21. Weiss, C.; Karras, P.; Bernstein, A. Hexastore: Sextuple indexing for semantic Web data management. *Proc. VLDB Endow.* **2008**, *1*, 1008–1019.
22. Cunningham, H.; Maynard, D.; Bontcheva, K.; Tablan, V. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL), Philadelphia, PA, USA, 6–12 July 2002.

23. TermRaider term extraction tools. Available online: https://gate.ac.uk/sale/tao/splitch23.html#sec:creole:termraider (accessed on 23 October 2014).
24. Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D.; Mendes, P.N.; Hellmann, S.; Morsey, M.; van Kleef, P.; Auer, S.; Bizer, C. DBpedia—A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal* **2014**, doi:10.3233/SW-140134.
25. Bollacker, K.; Evans, C.; Paritosh, P.; Sturge, T.; Taylor, J. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In Proceedings of the ACM SIGMOD Conference (SIGMOD'08), Vancouver, Canada, 9–12 June 2008.
26. Nunes, B.P.; Dietze, S.; Casanova, M.; Kawase, R.; Fetahu, B.; Nejdl, W. Combining a co-occurrence-based and a semantic measure for entity linking. In Proceeedings of the 10th Extended Semantic Web Conference (ESWC), Montpellier, France, 26–30 May 2013.
27. Demidova, E.; Oelze, I.; Nejdl, W. Aligning Freebase with the YAGO Ontology. In Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management (CIKM), San Francisco, CA, USA, 27 October–1 November 2013.
28. Suchanek, F.M.; Kasneci, G.; Weikum, G. Yago: A Core of Semantic Knowledge. In Proceedings of the 16th International World Wide Web Conference, Banff, Alberta, Canada, 8–12 May 2007.
29. Poblete, B.; Gavilanes, R.O.G.; Mendoza, M.; Jaimes, A. Do all birds tweet the same?: Characterizing Twitter around the world. In Proceedings of the 20th ACM Conference on Information and Knowledge Management (CIKM), Glasgow, UK, 24–28 October 2011.
30. Siehndel, P.; Kawase, R. TwikiMe!—User Profiles That Make Sense. In Proceedings of the ISWC 2012 Posters & Demonstrations Track, Boston, MA, USA, 11–15 November 2012.
31. Kawase, R.; Siehndel, P.; Pereira Nunes, B.; Herder, E.; Nejdl, W. Exploiting the wisdom of the crowds for characterizing and connecting heterogeneous resources. In Proceedings of the 25th ACM Conference on Hypertext and Social Media, Santiago, Chile, 1–4 September 2014.
32. Wikipedia Miner Toolkit. Available online: http://wikipedia-miner.cms.waikato.ac.nz/ (accessed on 23 October 2014).
33. Milne, D.; Witten, I.H. An open-source toolkit for mining Wikipedia. *Artif. Intell.* **2013**, *194*, 222–239.
34. Kwak, H.; Lee, C.; Park, H.; Moon, S. What is Twitter, a Social Network or a News Media? In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC, USA, 26–30 April 2010.
35. Stajner, T.; Thomee, B.; Popescu, A.M.; Pennacchiotti, M.; Jaimes, A. Automatic selection of social media responses to news. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), Chicago, IL, USA, 11–14 August 2013.
36. Maniu, S.; Cautis, B. Taagle: Efficient, personalized search in collaborative tagging networks. In Procedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD), Scottsdale, AZ, USA, 20–24 May 2012.
37. Maniu, S.; Cautis, B. Efficient Top-K Retrieval in Online Social Tagging Networks. *CoRR* **2011**, *abs/1104.1605*. Available online: http://arxiv.org/abs/1104.1605 (accessed on 23 October 2014).

38. Tahmasebi, N.; Gossen, G.; Kanhabua, N.; Holzmann, H.; Risse, T. NEER: An Unsupervised Method for Named Entity Evolution Recognition. In Proceedings of the 24th International Conference on Computational Linguistics (COLING), Mumbai, India, 8–15 December 2012.
39. Tahmasebi, N.; Niklas, K.; Theuerkauf, T.; Risse, T. Using Word Sense Discrimination on Historic Document Collections. In Proceedings of the 10th ACM/IEEE Joint Conference on Digital Libraries (JCDL), Gold Coast, Queensland, Australia, 21–25 June 2010.
40. O'Connor, B.; Balasubramanyan, R.; Routledge, B.R.; Smith, N.A. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. In Proceedings of the 4th International Conference on Weblogs and Social Media (ICWSM), Washington, DC, USA, 23–26 May 2010.
41. Jiang, L.; Yu, M.; Zhou, M.; Liu, X.; Zhao, T. Target-dependent Twitter Sentiment Classification. In Proceedingsa of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011.
42. Plachouras, V.; Stavrakas, Y. Querying Term Associations and their Temporal Evolution in Social Data. VLDB Workshop on Online Social Systems (WOSS), 31 August 2012, Istanbul, Turkey.
43. Mohr, G.; Stack, M.; Ranitovic, I.; Avery, D.; Kimpton, M. Introduction to heritrix, an archival quality Web crawler. In Proceedings of the 4th International Web Archiving Workshop, Bath, UK, 16 September 2004.
44. Plachouras, V.; Carpentier, F.; Faheem, M.; Masanès, J.; Risse, T.; Senellart, P.; Siehndel, P.; Stavrakas, Y. ARCOMEM Crawling Architecture. *Future Internet* **2014**, *6*, 518–541.
45. Arvidson, A.; Lettenström, F. The Kulturarw Project—The Swedish Royal Web Archive. *Electron. Libr.* **1998**, *16*, 105–108.
46. International Internet Preservation Consortium (IIPC). Available online: http://netpreserve.org/ (accessed on 23 October 2014).
47. Masanès, J. *Web Archiving*; Springer: Berlin, Germany, 2006; pp. I–VII, 1–234.
48. Living Web Archives Project. Available online: http://www.liwa-project.eu/ (accessed on 23 October 2014).
49. Cho, J.; Garcia-Molina, H.; Page, L. Efficient crawling through URL ordering. In Proceedings of the 7th International Conference on World Wide Web, Brisbane, Australia, 14–18 April 1998.
50. Baeza-Yates, R.; Castillo, C.; Marin, M.; Rodriguez, A. Crawling a Country: Better Strategies Than Breadth-first for Web Page Ordering. In Proceedings of Special Interest Tracks and Posters of the 14th International Conference on World Wide Web, Chiba, Japan, 10–14 May 2005.
51. Chakrabarti, S.; van den Berg, M.; Dom, B. Focused Crawling: A New Approach to Topic-specific Web Resource Discovery. *Comput. Netw.* **1999**, *31*, 1623–1640.
52. Menczer, F.; Pant, G.; Srinivasan, P. Topical Web crawlers: Evaluating adaptive algorithms. *ACM Trans. Internet Technol.* **2004**, *4*, 378–419.
53. Laranjeira, B.; Moreira, V.; Villavicencio, A.; Ramisch, C.; Finatto, M.J. Comparing the Quality of Focused Crawlers and of the Translation Resources Obtained from them. In Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC), Reykjavik, Iceland, 26–31 May 2014.
54. Pandey, S.; Olston, C. User-centric Web Crawling. In Proceedings of the 14th International Conference on World Wide Web, Chiba, Japan, 10–14 May 2005.

55. Barford, P.; Canadi, I.; Krushevskaja, D.; Ma, Q.; Muthukrishnan, S. Adscape: Harvesting and Analyzing Online Display Ads. In Proceedings of the 23rd International Conference on World Wide Web, Seoul, Korea, 7–11 April 2014.
56. Boanjak, M.; Oliveira, E.; Martins, J.; Mendes Rodrigues, E.; Sarmento, L. TwitterEcho: A Distributed Focused Crawler to Support Open Research with Twitter Data. In Proceedings of the Workshop on Social Media Applications in News and Entertainment (SMANE 2012), at the ACM 2012 International World Wide Web Conference, Lyon, France, 16 April 2012.
57. Psallidas, F.; Ntoulas, A.; Delis, A. Soc Web: Efficient Monitoring of Social Network Activities. In Proceedings of the 14th International Web Information Systems Engineering Conference, Nanjing, China, 13–15 October 2013.
58. Blackburn, J.; Iamnitchi., A. An architecture for collecting longitudinal social data. In IEEE ICC Workshop on Beyond Social Networks: Collective Awareness, Budapest, Hungary, 9 June 2013.
59. Isele, R.; Umbrich, J.; Bizer, C.; Harth, A. LDSpider: An open-source crawling framework for the Web of Linked Data. In Proceedings of 9th International Semantic Web Conference (ISWC) Posters and Demos, Shanghai, China, 9 November 2010.
60. Slug: A Semantic Web Crawler. Available online: http://www.ldodds.com/projects/slug/ (accessed on 23 October 2014).
61. Internet Archive. NutchWAX. Available online: http://archive-access.sourceforge.net/projects/nutch/ (accessed on 30 January 2012).
62. Apache Nutch—Highly extensible, highly scalable Web crawler. Available online: http://nutch.apache.org/ (accessed on 23 October 2014).
63. Gomes, D.; Cruz, D.; Miranda, J.A.; Costa, M.; Fontes, S.A. Search the Past with the Portuguese Web Archive. In Proceedings of the 22nd International Conference on World Wide Web (Companion Volume), Rio de Janeiro, Brazil, 13–17 May 2013.
64. Internet Archive. Wayback. Available online: http://archive-access.sourceforge.net/projects/wayback/ (accessed on 30 January 2012).
65. Spaniol, M.; Weikum, G. Tracking Entities in Web Archives: The LAWA Project. In Proceedings of the 21st International Conference Companion on World Wide Web (Companion Volume), Lyon, France, 16–20 April 2012.
66. Big UK Domain Data for the Arts and Humanities. Available online: http://buddah.projects.history.ac.uk/ (accessed on 23 October 2014).