Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 9, Issue, 10, pp.59162-59171, October, 2017

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

ISSN: 0975-833X

RESEARCH ARTICLE

A SEMANTIC WEB METADATA HARVESTING AND ENRICHMENT MODEL FOR DIGITAL LIBRARY AND SOCIAL NETWORKS

Ronald Brisebois, *Apollinaire Nadembega, Philippe N'techobo and Herve Landry Djeuteu

Bibliomondo, Montréal, Canada

ARTICLE INFO

Article History:
Received 28th July, 2017
Received in revised form
13th August, 2017
Accepted 27th September, 2017
Published online 31st October, 2017

Key words:

Metadata harvesting,
Metadata interoperability,
Semantic digital library,
Semantic metadata enrichment,
Semantic topic detection,
Sentiment analysis.

ABSTRACT

Web semantic metadata rules-based harvesting became is an important challenge due to validation of the semantic metadata and the amount of web sites that are rich knowledge sources. Indeed, extracting useful information from the web is the most significant issue of concern for the realization of semantic web; this may be achieved by several ways among which web usage mining, web crawling and scrapping and semantic annotation plays an important role. In this paper, a semantic web metadata harvesting and enrichment model, called Semantic Universal Knowledge Model (SUKM). It goal is to make an enriched semantic encyclopedia. SUKM has to support multi-platform metadata driven applications and interoperability. It may be defined as a structure and rich version of DBpedia in order to increase the usability of various user web knowledge experiences. SUKM aggregates and enriches metadata to create a semantic master metadata catalogue. More specifically, a harvesting modelconsisting of five phases is proposed. This model takes into account sources classification, type of source contents and semantic relationships. SUKM model includes metadata cleaning to remove duplication from different source and semantic metadata enrichments. Semantic Metadata Enrichments consist to identify and enrich topic and emotion metadata hidden within the text or multimedia structure. Enrichment processes use a hybrid machine learning model to propose a topic detection and emotion analysis algorithms. SUKM rules-based harvesting prototype has been implemented using a Java program and more than 10 million metadata hybrid documents have been integrated to the semantic master metadata catalogue.

Citation*:* Ronald Brisebois, Apollinaire Nadembega, Philippe N'techobo and Herve Landry Djeuteu, 2017. "A semantic web metadata harvesting and enrichment model for digital library and social networks", *International Journal of Current Research*, 9, (10), 59162-59171.

INTRODUCTION

The size of web is tremendously large and continue to grow, how to harvest continuously relevant information from multiple sources and to keep integrity of the knowledge? It is impossible to manually analyze all the information contains in the web. In addition, web databases generate query result pages based on a user's query; automatically extracting the data from these query result pages is very important for many applications, such as data integration, which need to cooperate with multiple web databases. To overcome this issue, one solution is to use semantic metadata analysis and rules-based harvesting to enhance the harvesting extraction processes. Metadata is structured information that describes, explains, locates, accesses, retrieves, uses, or manages an information resource of any kind; metadata is often, called data about data or information about information. Semantic Metadata refers to semantic relationship about data. Some use it to refer to machine understandable information, while others employ it only for records that describe electronic resources.

Corresponding author: **Apollinaire Nadembega**
Bibliomondo, Montréal, Canada.

In the digital library ecosystem, metadata is commonly used for any formal scheme of resource description, applying to any type of object, digital or non-digital. Many metadata schemes exist to describe various types of textual and non-textual objects as digital library including published book, movie and video, electronic document, archival document, social network, art object, educational and training material, scientific datasets and, obviously, the web.Once the web databases are structured using metadata, these metadata must be centralized in the one repository; that is the raison of metadata harvesting process. The essence of harvesting is to enable access to web-accessible material through inter operable repositories for metadata sharing, publishing and archiving. The sharing of knowledge may lead to further development in to the same discipline or related discipline. To obtain central enriched metadata repository, Metadata harvesting from several and different metadata sources need to be performed. Metadata harvesting is a technique for gathering together of metadata from a number of distributed repositories into a combined data store. For the semantic metadata harvesting the combined data store become a combined triplets, called RDF triplestore; RDF triplestore is a type of graph database that stores semantic facts. The data in RDF triplestore is stored in the relationship: Subject, Predicate

and Object; for example, "Bob is 35" or "Bob knows Fred". Metadata harvesting technique consists of web crawling and data scraping. Web crawling can be defined as web page url gathering while web data scraping is defined as web page metadata gathering. Websites crawling can be achieved using many crawling frameworks, such as scrapy for Python. Unfortunately, such frameworks that traverse the links of websites need to be tailored to the specific use case. In addition, data scraping and information extracting from sources is required to convert the raw data that the crawler retrieves into a format that is suitable for further analysis tasks, such as natural language processing. In the literature, two main approaches of metadata harvesting are presented: (a) Open Archive Initiative Protocol for Metadata Harvesting (OAI-PMH) and (b) HTML Web page which is more complex due to the uniqueness of each HTML page. According to the literature, one of metadata harvesting for central repository is the metadata disambiguation and duplication. For metadata disambiguation, specific thesauri are used in library domain while for metadata duplication, only duplication algorithm needs to be defined. OAI-PMH specifies two players in the harvesting process- Data provider, who create structural metadata and expose them for harvesting and Service provider, who harvest and normalize the structured metadata, providing a searchable interface to search for and retrieve metadata records. The harvesting process is consisting of the service provider using HTTP to request information from the data provider, which responds in accordance with the established protocol. In order to take into account any type of metadata sources, our proposal includes the both metadata-harvesting approaches.

In this paper, we present a novel metadata harvesting and enrichment model, called Semantic Universal Knowledge Model (SUKM), to support semantic metadata enrichments driven applications or API. The goal of SUKM is to allow multi-sources and multi-contents types harvesting and enrichment in order to provide a semantic master metadata catalogue (SMMC) as a rich semantic encyclopedia of knowledgeable metadata. More specifically, (i) SUKM, first, identifies the type of metadata sources and classifies them according to their relevance for specific subject or field; this phase is called *"Sources analysis"*. After sources analysis phase, (ii) SUKM harvests the link to the website of each relevant metadata sources; this phase is called *"Links harvesting"*. Once links to notices harvesting phase performed, (iii) SUKM harvests the metadata of contents located into the website of each relevant metadata sources; this phase is called *"Semantic metadata harvesting"*. The next phase of SUKM is (iv) the metadata deduplication and merging; this phase is called *"Metadata cleaning"*. After Metadata cleaning phase, (v) SUKM can download digital documents related to each notice without worrying about the unnecessary use of storage space; this phase is called *"Documents downloading"*. Finally, (vi) SUKM performs internal enrichment based on digital documents analysis (Brisebois, 2017 and Brisebois, 2017), this phase is called *"Metadataenrichments"*. SUKM and SMMC universal repository is our semantic metadata enrichment software ecosystem (SMESE) (Brisebois, 2017). The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 briefly describes a semantic metadata enrichment software ecosystem (SMESE) while Section 4 describes the metadata harvesting and enrichment model, called Semantic Universal Knowledge Model (SUKM) and its various components. Section 5 presents the evaluation through a number of simulations. Section 6 presents a summary and some suggestions for future work.

Related work

This paper focuses on the issue of providing semantic master metadata catalogue (SMMC) based on semantic web metadata harvesting and enrichment model, called Semantic Universal Knowledge Model (SUKM); this contribution is at the intersection of SMESE and web metadata harvesting and semantic metadata enrichment. The related work section coverts these research field.

Semantic metadata enrichment software ecosystem (SMESE)

The development of SMESE consists of Software product line engineering (SPLE) and SECO architecture. This involves requirements analysis, design, construction, testing, configuration management, quality assurance and more, where stakeholders always look for high productivity, low cost and low maintenance. This has led to software product line engineering (SPLE) (Capilla, 2014; Olyai, 2015; Horcas, 2016 and Ayala, 2015), and software ecosystems (SECO) [8-10] as a comprehensive model that helps software providers to build applications for organizations/clients based on a common architecture and core assets.SPLE is a set of software intensive systems that share a common and managed set of features satisfying the specific needs of a particular market segment developed from a common set of core assets in a prescribed way (Olyai, 2015). SPL engineering aims at: effective utilization of software assets, reducing the time required to deliver a product, improving quality, and decreasing the cost of software products. SPLE deals with the assembly of components from a component-based architecture (He, 2014), and involves the continuous growth of the number of components. An overview of SPLE challenges is presented in (Capilla, 2014). In literature, three trends expected from SPLE research are identified: (1) Managing variability in non-product-line settings; (2) Leveraging instantaneous feedback from big data and (3) Addressing the open world and open functionalities assumption in software product line settings. A survey of works on search based software engineering (SBSE) for SPLE is presented in (Capilla, 2014). SECO consist of multiple software projects, often interrelated to each other by means of dependency relationships. When one project undergoes changes and issues a new release, this may or may not lead other projects to upgrade their dependencies. Unfortunately, the upgrade of a component may create a series of issues. In their systematic literature review of SECO research, Manikas and Hansen (Manikas, 2013), report that while research on SECO is increasing: (1) There is little consensus on what constitutes a SECO; (2) Few analytical models of SECO exist and (3) Little research is done in the context of real-world SECO.Some studies (Demir, 2015; Neves, 2014; Alférez, 2014; Singh, 2015 and Yadav, 2015), focused on SECO architecture related to SPLE, beginning with an industry perspective. Christensen *et al*. (Christensen, 2014), define the concept of SECO architecture as a set of structures comprised of actors and software elements, the relationships among them, and their properties. Neves *et al*. (Neves, 2014), propose an architectural solution based on ontology and the spreading algorithm that offers personalized and contextualized event recommendations while Alferez *et al*. (Alférez, 2014), propose a framework that uses semantically

rich variability models at runtime to support the dynamic adaptation of service compositions. To include component based software development (CBSD) in SECO, the fuzzy logic approach (Singh, 2015 and Yadav, 2015), is largely used to select components.

Metadata harvesting and enrichments

Interest in entity metadata harvesting was initially limited to those in the community who preferred to concentrate on manual design of ontologies as a measure of quality. Following the linked data bootstrapping provided by DBpedia, many changes ensued with a related need for substantial population of knowledge bases, schema induction from data, natural language access to structured data, and in general all applications that make for joint exploitation of structured and unstructured content. In practice, Graph-based methods, meanwhile, were incrementally entering the toolbox of semantic technologies at large. Metadata harvesting (Yadav, 2015 and Sufyan, 2016), and their enrichment is the core engine of semantic master metadata catalogue (SMMC). Metadata harvesting and enrichments consist of:

- Semantic metadata harvesting,
- Semantic topic detection (STD),
- Sentiment and emotion analysis (SEA).

Several surveys (Kadam, 2014; Patel, 2015; Sufyan, 2016; Ferrara, 2014Dastidar, 2016), are been presented in literature about Metadata harvesting. In the literature, two main approaches of metadata harvesting are presented:

- Open Archive Initiative Protocol for Metadata Harvesting (OAI-PMH),
- HTML Web page.

The OAI-PMH (Kumar Roy, 2017 and Kapidakis, 2015), is a protocol designed to harvest metadata records, so that they can be collected from multiple sources (repositories) and processed in one place. It transfers selected records, normally only the additional or changed ones since the last transfer. The metadata transfer can be quite efficient as the record format rarely change, because a format change usually requires manual intervention. Each OAI-PMH communication needs agents in two roles: data providers that act as OAI-PMH servers to provide their metadata records or any other (mostly structured) data that they wish to share and harvesters that act as OAI-PMH clients to retrieve records from the data providers and feed them to the receiving applications (e.g. metadata aggregators). HTML Web page metadata harvesting is more complex than OAI-PMH. Indeed, few web metadata harvesting engine are proposed based on HTML. Generally, a topic is represented as a set of descriptive and collocated keywords/ terms. Initially, document clustering techniques were adopted to cluster content-similar documents and extract keywords from clustered document sets as the representation of topics (subjects). The predominant method for topic detection is the latent Dirichlet allocation (LDA) (Blei, 2003), which assumes a generating process for the documents. LDA has been proven a powerful algorithm because of its ability to mine semantic information from text data. Terms having semantic relations with each other are collected as a topic. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. The literature presents two groups of text-based topic detection approaches based on the size of the text: short text (Dang, 2016; Cigarrán, 2016; Cotelo, 2016), such as tweets or Facebook posts, and long text (Blei, 2003; Zhang, 2016; Chen, 2016) such as a book. For example, Dang *et al*. (2016). proposed an early detection method for emerging topics based on dynamic Bayesian networks in micro-blogging networks while Cigarran *et al*. (Cigarrán, 2016), proposed an approach based on formal concept analysis (FCA). Many of topic detection techniques (Zhang, 2016; Chen, 2016)rely heavily on simple keyword extraction from text.The main objective of SEA is to establish the attitude of a given person with regard to sentences, paragraphs, chapters or documents (Appel, 2016; Patel, 2016; Balazs, 2016; Ravi, 2015Serrano-Guerrero, 2015; Vilares, 2015 and Kiritchenko, 2014). In addition, with the rapid spread of social media, it has become necessary to categorize these reviews in an automated way. There are three main techniques for SEA: keyword spotting, lexical affinity and statistical methods. The first two methods are well known while statistical methods have to be more explored further. Statistical methods, such as Bayesian inference and support vector machines, are supervised approaches in which a labeled corpus is used for training a classification method which builds a classification model used for predicting the polarity of novel texts. By feeding a large training corpus of affectively annotated texts to a machine learning algorithm, it is possible for the system to not only learn the affective valence of related keywords (as in the keyword spotting approach), but also to take into account the valence of other arbitrary keywords, punctuation, and word co-occurrence frequencies. Sentiment analysis can be carried out at different levels of text granularity: document (Cho, 2014), sentence (Appel, 2016; Patel, 2016 and Desmet, 2013), phrase, clause, and word (Quan, 2014). Sentiment analysis may be at the sentence or phrase level (which has recently received quite a bit of research attention) or at the document level. Emotions are also associated with mood, temperament, personality, outlook and motivation (Munezero, 2014). However, sentiments are differentiated from emotions by the duration in which they are experienced. The SWAT model was proposed to explore the connection between the evoked emotions of readers and news headlines by generating a word-emotion mapping dictionary. For each word w in the corpus, it assigns a weight for each emotion e, i.e., P(e|w) is the averaged emotion score observed in each news headline H in which w appears. The emotion-term model is a variant of the NB classifier and was designed to model word-emotion associations. In this model, the probability of word wj conditioned on emotion ek is estimated based on the co-occurrence count between word wj and emotion ek for all documents. The emotion-topic model is combination of the emotion-term model and LDA. Cambria *et al*. (Cambria, 2015) explored how the high generalization performance, low computational complexity, and fast learning speed of extreme learning machines can be exploited to perform analogical reasoning in a vector space model of affective common-sense knowledge.

SMESE multi-platform architecture interoperability

This section presents the semantic enriched metadata software ecosystem (SMESE) architecture based on SPLE and SECO approaches to support semantic metadata enrichment for digital libraries interoperability [3]. The SMESE multiplatform prototype includes an engine to aggregate multiple world

catalogues. The SMESE multiplatform framework must link bibliographic records and semantic metadata enrichments. Semantic relationships between the content, person, organization and places are defined and curated in the master metadata catalogue. Topics, sentiment and emotions must be extracted automatically from the contents and their semantic context:

- Libraries spend a lot of money buying books and electronic resources. Enrichment uncovers that information and makes it possible for people to discover the great resources available everywhere.
- The average library has hundreds of thousands of catalogue records waiting to be transformed into linked data, turning those thousands of records into millions of relationships and triplets.
- FRBR (functional requirements for bibliographic records) is a semantic representation of the bibliographic record. A work is a high-level description of a document, containing information such as author (person), title, descriptions, subjects, etc., common to all expressions, format and copy of the work. (See Fig. 1 for an FRBR framework description).

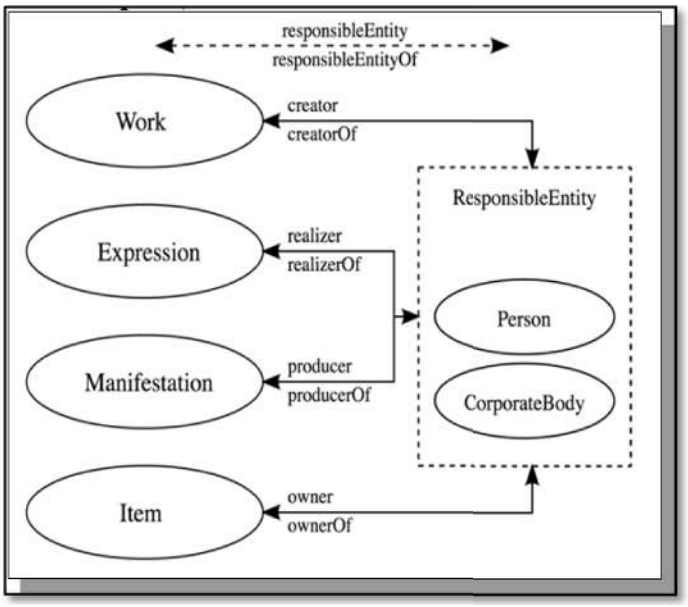

Fig. 1. FRBR framework description

This new semantic ecosystem will harvest and enrich bibliographic records externally (from the web) and internally (from text data). The main components of the ecosystem will be:

- Metadata initiatives & concordance rules,
- Harvesting web metadata & data,
- Harvesting authority metadata & data,
- Rule-based semantic metadata external enrichment engine,
- Rule-based semantic metadata internal enrichment engine,
- Semantic metadata external & internal enrichment synchronization engine,
- User interest-based gateway,
- Semantic master catalogue.

Fig. 2 shows the SMESE architecture; the key elements of SMESE are:

- A software ecosystem model that configures the application production process including software aspects based on a proposed CBSD and metadata-based SPLE approach.
- An implementation of semantic metadata enrichment using SPLE and a semantic master metadata catalogue for a semantic digital library.

A.Metadata initiatives & concordance rules

Several rules have been proposed to cover the description and provision of access points for all library materials (entities). These rules are based on an individual framework for the description of library documents and their semantic relationships. Here, we proposed a unified interoperable model between most known metadata models: Dublin Core (DC), UNIMARC, MARC21, RDF/RDA and BIBFRAME.

B.Harvesting of web metadata & data

The harvesting of web metadata & data sources such as: (1) Semantic digital resources, (2) Digital resources, (3) Portal/websites events, (4) Social networks & events, (5) Enrichment repositories, and (6) Discovery repositories. The integration of these sources in SMESE allows users to aggregate and enrich metadata and data.

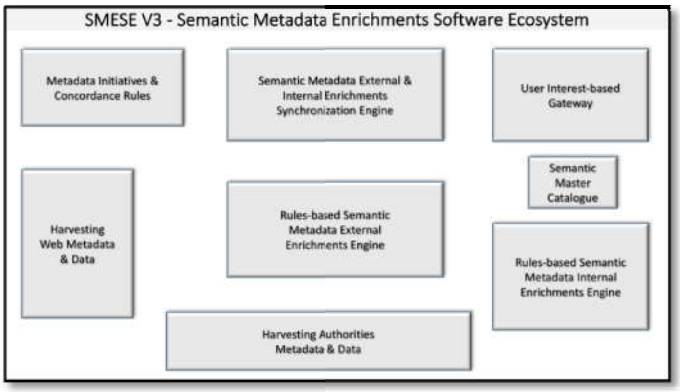

Fig. 2. Semantic Enriched Metadata Software Ecosystem (SMESE) Architecture

C.Harvesting authority metadata & data

This bloc represents the details of the Harvesting of Authorities Metadata & Data. The Semantic Multi-Platform Ecosystem consists of many authority sources, such as: (1) BAnQ - Bibliothèque et Archives nationales du Qc, (2) Bibliothèque et Archives du Canada, (3) Bibliothèque Nationale de France, (4) Library of Congress, (5) British Library, and (6) Europeana.

D.Rules-based semantic metadata external enrichments engine

This bloc represents the details of the rule-based semantic metadata external enrichment engine. Semantic searches over documents and other content types needs to use semantic metadata enrichment (SME) to find information based not just on the presence of words, but also on their meaning. It consists of: (1) Rule-based semantic metadata external enrichment engine, (2) Multilingual normalization, (3) Rule-based data conversion and (4) Harvesting metadata & data.

E.Rule-based semantic metadata internal enrichments engine

This bloc represents the details of the rule-based semantic metadata internal enrichment engine including software product line engineering (SPLE). This sub-system includes: (1) A rule-based semantic metadata internal enrichment engine, (2) A multilingual normalization process, (3) Software Product Line Engineering (SPLE) and (4) A topic, sentiment/emotion, abstract analysis and an automatic literature review.

F.Semantic metadata external & internal enrichments synchronization engine

This bloc represents the semantic metadata external & internal enrichment synchronization engine. This engine identifies which processes to synchronize and which semantic enrichments to push outside the ecosystem.

G.User interest-based gateway

This bloc represents the user interest-based gateway (UIG) that represents the person (mobile or stationary) who interacts with the ecosystem.The users and contributors are categorized into five groups: (1) Interest-based gateway (mobile-first), (2) Semantic Search Engine (SSE), (3) Discovery, (4) Notifications and (5) Metadata source selection.

H.Semantic master catalogue

This bloc represents the semantic master catalogue (SMC) that represents the knowledge base of the SMESE ecosystem.

Semantic Universal Knowledge Model (SUKM)

This section presents the semantic universal knowledge model (SUKM). SUKM is composed to 6 mains phases:

- Sources analysis,
- Links harvesting,
- Semantic metadata harvesting,
- Metadata cleaning,
- Documents downloading,
- Metadata enrichment.

In the following, we describe SUKM processes in detail.

SUKM links to contents notice harvesting process

SUKM starts by a manual task, called sources analysis, whose goal is to identify relevant sources for specific contents. For example, for the contents related to the music, Discogs (www.discogs.com) and ALLMUSIC (www.allmusic.com) are identified while for scientific publications, Sciencedirect (www.sciencedirect.com) and Researchgate (www. researchgate.net) are used. In addition to classification task, sources analysis consists also to define the type of harvesting technique that needs to be applied and its complexity level. The output of the sources analysis is the table with the following columns:

- Name,
- Website or social network,
- Content type,
- Amount of content,
- Harvesting technique,
- Harvesting complexity level,
- Metadata model.

After sources analysis, the next phase of SUKM is the links harvesting that consists to harvest the url to access to the metadata of content. Specifically, SUKM generates a hierarchical tree model for each source where the node denotes the url to access to a specific web page; each node is identified by its index; we assume that the root has index zero. The navigation model in the source website defines the link between the nodes. In the generated hierarchical tree, the leaf nodes denote the url to the metadata of content and sources. The advantage of using tree model is the fact that it avoids to restart the harvesting at the beginning in case of fails. Fig. 3 illustrates the sources analysis and links harvesting phases.

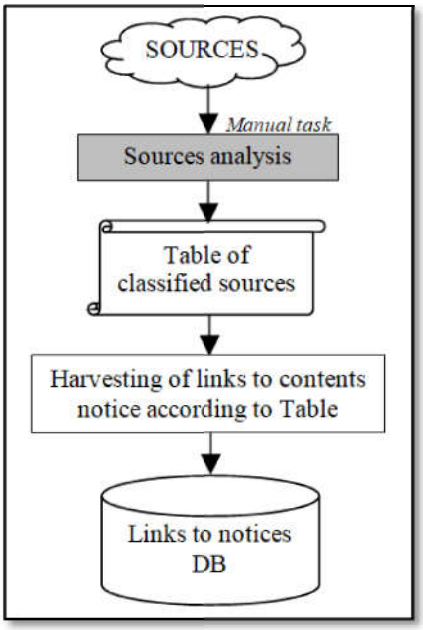

Fig. 3. SUKM sources analysis and links harvesting workflow

SUKM semantic metadata harvesting process

Here, the third phase of SUKM is presented that goal is to harvest the metadata of contents located into the website of each relevant metadata sources. Indeed, using the second phase output, SUKM performs metadata harvesting based on the hierarchical trees of the sources. Each hierarchical tree is characterized by its depth *d* and its number of leaf nodes *n*. Let $H_i^T(S_i, d_i, n_i)$ be the source *i*hierarchical tree of depth d_i with $n_i$leaf nodes; to facilitate understanding, this tree is called *original tree*. In order to perform efficient harvesting, an ongoing harvesting tree is built during the source harvesting; once one node of hierarchical tree is visited, SUKM creates this node in the ongoing harvesting tree related to this source. Let $O_i^T(S_i, l_i, m_i)$ be the source *i*ongoing hierarchical tree of depth l_i with $m_i$leaf nodes; in order to facilitate understanding, this tree is called *ongoing tree*.As shown in Fig. 4, SUKM harvesting phase may use several harvesting agents per source according to the size and the relative importance of the source. The number of harvesting agents for the source *i* is computed as follows:

$$\omega = A\left(H_i^T\right) = \frac{n_i}{d_i} \quad (1)$$

Thus $H_i^T(S_i,d_i,n_i)$ is splitted into ω sub trees where each sub tree is assigned to a harvesting agent which ran the harvesting algorithm. The task manager 1 is responsible for sub trees definition and assignation to harvesting agents. In order to carry out a balanced distribution of the workload, task manager removes part of tasks from overloaded agents and re-assigns them to agents which are completed their sub trees semantic harvesting. The pseudo-code of the semantic harvesting algorithm is presented below (see Table 1).

Table 1. Harvesting algorithm

Pseudo code: Harvesting algorithm
For (j=0;j++;j<n_i) // n_i is the $H_i^T(S_i,d_i,n_i)$ number of leaf nodes
a=Get (H_i^T ,j) // allow to get the root of $H_i^T(S_i,d_i,n_i)$
Harvest leaf node *a* metadata
Set(O_i^T , j, a) // add leaf node *a* into $O_i^T(S_i,l_i,m_i)$

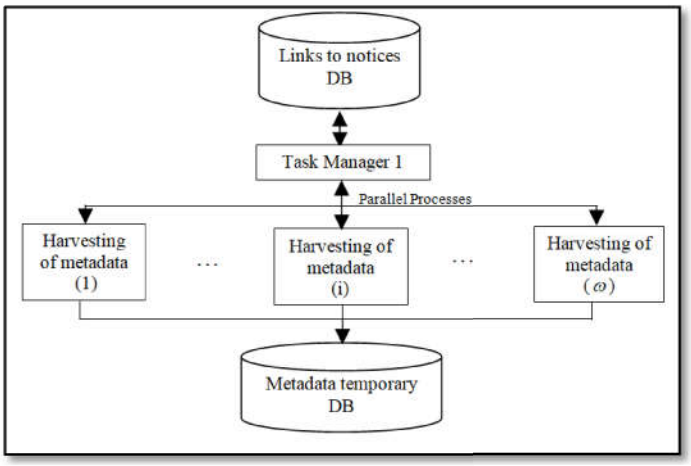

Fig. 4. SUKM metadata harvesting workflow

In case of abrupt stop due to error, SUKM does not restart the harvesting process from the beginning. SUKM makes use of re-harvesting procedure based on the pseudo-code of the harvesting algorithm presented below (see Table 2).

Table 2. Tree Matching algorithm

Pseudo code: Tree Matching algorithm
For (j=0;j++;j<N) // N is the $H_i^T(S_i,d_i,n_i)$ number of nodes
a=Get (H_i^T ,j) // allow to get the root of $H_i^T(S_i,d_i,n_i)$
b=Get (O_i^T ,j) // allow to get the root of $O_i^T(S_i,l_i,m_i)$
IF a =! b // check if u is into $O_i^T(S_i,l_i,m_i)$
Re-harvesting point = b
Call "Harvesting algorithm" starting to "Re-harvesting point"

SUKM metadata cleaning process

Fig. 5 illustrates the process applied by SUKM to perform the metadata deduplication and merging, called *"Metadata cleaning"*. For each entry of metadata temporary DB, task manager 2 identifies the content type of this entry. Then, according to the content type, task manager 2 identifies the set of cleaning agents are able to perform the task; indeed, each set of cleaning agents is pre-assigned to specific set of contents types. Let *m* be the number of cleaning agents required for an entry *e*. Each cleaning agent is responsible to check the presence of entry *e* into an assigned sub set of the "Deduplicated and Merged Metadata DB" (see Fig. 5). When one cleaning agent detected the presence of entry *e* into its sub set of "Deduplicated and Merged Metadata DB", it notifies the task manager 2 which asks the other cleaning agent to stop checking process.

If the entry *e* is not find in the "Deduplicated and Merged Metadata DB", task manager 2 is responsible to insert notice related to entry *e* into"Deduplicated and Merged Metadata DB".

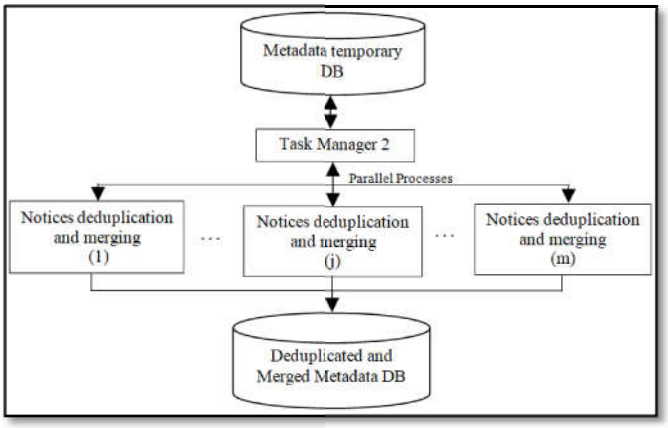

Fig. 5. SUKM metadata cleaning workflow

However, if the entry *e* is found in the "Deduplicated and Merged Metadata DB", task manager 2 performs merging process in order to add missing metadata to notice in "Deduplicated and Merged Metadata DB" related to entry *e*.

Documents downloading process

After Metadata cleaning phase, SUKM can download digital documents related to each notice without worrying about the unnecessary use of storage space (See Fig. 6).

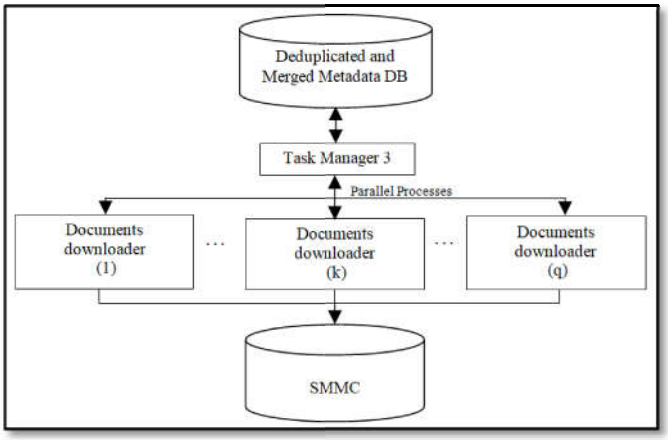

Fig. 6. SUKM document downloading workflow

This phase is called *"Documents downloading"*. Notice that the "Documents downloading" is a type of metadata harvesting. However, the ideal behind the separation of the document downloading and the other metadata harvesting is to avoid several duplications of the same documents and to keep metadata about all different sources of documents.Indeed, the duplication of notices may cause several downloading of the

same digital document;which should require more storage space and processing time; that is the main reason that the digital documents downloading are not performed during the metadata harvesting phase.Notice that, digital documents are not kept for a long time. They are just used to perform internal enrichment process. After enrichment, the digital documents are removed but all references are kept in the metadata catalogue.

SUKM metadata enrichment process

In this section, the semantic metadata internal enrichment (SMIE) process is presented. SMIE consists of two algorithms: (a) semantic-based topic detection (STD) algorithm and mood discovery in documents (MDD) algorithm. SMIE process through text analysis approaches for topics, sentiment/emotion and semantic relationships detection. To implement the STD and MDD algorithms, machine learning models have been used to perform metadata enrichments.

Semantic-based topic detection algorithm (STD)

The aim of STD is to build a classifier that can learn from already annotated contents (e.g., documents and books) and infer the topics of new books. Traditional approaches are typically based on various topic models, such as latent Dirichl*et al*location (LDA) where authors cluster terms into a topic by mining semantic relations between terms. However, co-occurrence relations across the document are commonly neglected, which leads to detection of incomplete information. Furthermore, the inability to discover latent co-occurrence relations via the context or other bridge terms prevents important but rare topics from being detected. STD combines semantic relations between terms and co-occurrence relations across the document making use of document annotation. In addition, STD includes:

- A probabilistic topic detection approach, called semantic topic model (SemTopicMod).
- A clustering approach that is an extension of KeyGraph, called semantic graph (SemGraph).

STD is a hybrid relation analysis and machine learning approach that integrates semantic relations, semantic annotations and co-occurrence relations for topic detection. More specifically, STD fuses multiple relations into a term graph and detects topics from the graph using a graph analytical method. It can detect topics not only more effectively by combining mutually complementary relations, but it can also mine important rare topics by leveraging latent co-occurrence relations. STD is based on our previous work [1].

Mood discovery in documents (MDD) algorithm

The MDD goal is to classify the corpus of documents taking emotion into consideration, and to determine which sentiment it more likely belongs to. A document can be a distribution of emotion $p(e|d)e \in E$ and a distribution of sentiment $p(s|d)s \in S$. MDD is a hybrid approach that combines a keyword-based approach and a rule-based approach. MDD is applied at the basic word level and requires an emotional keyword dictionary that has keywords (emotion words) with corresponding emotion labels. To refine the detection, MDD develops various rules to identify emotion. Rules are defined using an affective lexicon that contains a list of lexemes annotated with their affect. The purpose of MDD is to identify positive and negative opinions and emotions.For affective text evaluation, MDD uses the SS-Tagger (a part-of-speech tagger) and the Stanford parser. The Stanford parser was selected because it is more tolerant of constructions that are not grammatically correct. This is useful for short sentences such as titles. MDD also uses several lexical resources that create the MDD knowledge base located in the thesaurus. The lexical resources used are: WordNet, WordNet-Affect, SentiWordNet and NRC emotion lexicon. WordNet is a semantic lexicon where words are grouped into sets of synonyms, called synsets. WordNet-Affect is a hierarchy of affective domain labels that can further annotate the synsets representing affective concepts.One of the main component of MDD is the thesaurus, called BM emotion word model (EmoWordMod). EmoWordMod is an emotion-topic model that provides the emotional score of each keyword by taking the topic into account. EmoWordMod introduces an additional layer (i.e., latent topic) into the emotion-term model such as SentiWordNet. MDD is composed of three phases: EmoWordModgeneration process phase, sentiment and emotion discovery process phase and third sentiment and emotion refining process phase. MDD is based on our previous work [2].

Evaluation using simulations

Here, we evaluate, via simulations, the performance of SUKM. As comparison terms, we use Scrapy, schemes described in [46]. Scrapy was selected because, to the best of our knowledge, it is the most known web metadata harvester. To measure SUKM and Scrapy performance, a simulator program has been developed using Java code. The server characteristics for the simulations were: Dell Inc. PowerEdge R630 with 96 Ghz (4 x Intel(R) Xeon(R) CPU E5-2640 v4 @ 2.40GHz, 10 core and 20 threads per CPU) and 256 GB memory running VMWare ESXi 6.0.

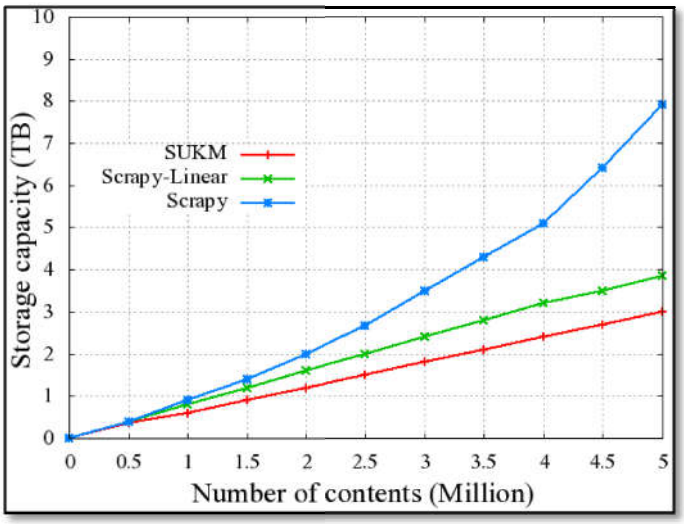

Fig. 7. Number of contents VS Storage capacity

Fig. 7 shows the required storage capacity for varying number of harvested contents, including the duplicated contents. In this set of simulations, three servers are used simultaneously for each prototype. In the figure, the Scrapy version in case of linear growth is referred to as Scrapy-Linear; notice that illustration of Scrapy-Linear is just to show the exponential growth of required storage capacity for regular Scrapy

prototype. It was observed that SUKM (red) outperforms Scrapy (blue). SUKM requires an average of 0.6 TB per 1 million harvested contents while Scrapy requires an average of 1.18 TB per 1 million harvested contents; the average relative improvement of SUKM compared with Scrapy is about 0.58 per 1 million harvested contents. This can be explained by the fact that SUKM removes the duplicated contents. Indeed, Scrapy does not perform metadata cleaning; then, it requires more storage capacity than SUKM. In addition, Scrapy-Linear compared with Scrapy shows that Scrapy harvests more than one copy for the same content.Fig. 8 shows the number of harvested contents when varying the number of servers. In this set of simulations; in this set of simulations, ten servers are used simultaneously for each prototype in order to reduce the harvesting time. Here, we observed that Scrapy (blue) outperforms SUKM (red) in terms of number of harvested contents. However, Scrapy harvested contents are not clean; that allows understanding that obtaining clean and quality metadata requires process time.

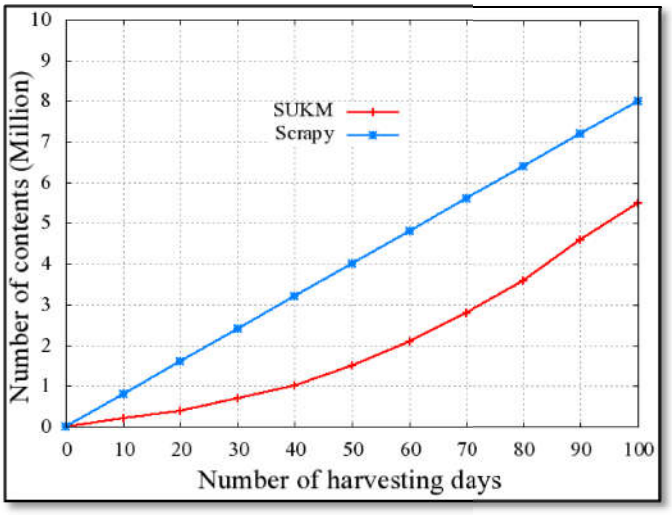

Fig. 8. Number of harvesting days VS Number of contents

One of the advantages of SUKM is the fact that it requires very few manual involvements; for example, in case of abrupt stop, it restarts alone. Table 3 presents the current number of harvested contents according to the contents types.

Table 3. Current number of harvested contents

Contents types	Number of harvested contents
Books	5 001 850
Scientific papers	3 499 796
TV programs	2 117 369
Totals	10 619 015

Summary and future work

In this paper, we have proposed a semantic web metadata harvesting and enrichment model, called Semantic Universal Knowledge Model (SUKM), with the goal to make an enriched semantic encyclopedia. SUKM is composed to six mains phases: (1) sources analysis that aims to identify relevant sources for specific contents and to define the type of harvesting technique that needs to be applied, (2) links harvesting that aims to harvest the url to access to the metadata of content and sources, (3) semantic metadata harvesting with the goal to harvest the metadata of contents located into the website of each relevant metadata sources, (4) metadata cleaning that aims to perform the metadata deduplication and merging, (5) documents downloading that aims to download digital documents related to each notice without and (6) metadata enrichment that consists of two machine learning algorithms: (a) semantic-based topic detection (STD) algorithm for topics detection and mood discovery in documents (MDD) algorithm for sentiment/emotion detection. The contribution of SUKM compared with Scrapy are:

- Each of the five phases can be run in parallel on different physical environments,
- The ease of maintenance due to the separation of components such as manager and database; indeed, the stopping of an agent does not prevent the others to continuing their tasks,
- The access to temporary db if content is not available in the deduplicated and merged db (see fig. 5),
- The significant reduction of connection session time; indeed, the link harvesting (phase 1) and metadata cleaning (phase 4) do not need connection to the sources; in contrast to a technique that integrates harvesting and cleaning which needs the login session, our approach avoids the risk of a interruption of session due to a long login session.

For the future work, it is planned to detail the harvesting process and improve the semantic protocol and algorithms.

REFERENCES

Alférez, G.H, Pelechano, V., Mazo, R., Salinesi, C., Diaz, D. 2014. Dynamic adaptation of service compositions with variability models. *Journal of Systems and Software* 91:24-47. doi:http://dx.doi.org/10.1016/j.jss.2013.06.034

Appel, O., Chiclana, F., Carter, J., Fujita, H. 2016. A hybrid approach to the sentiment analysis problem at the sentence level. Knowledge-Based Systems 108:110-124. doi:http://dx.doi.org/10.1016/j.knosys.2016.05.040

Ayala, I., Amor, M., Fuentes, L., Troya, J.M. 2015. A Software Product Line Process to Develop Agents for the IoT. Sensors 15 (7):15640-15660 doi:10.3390/s150715640

Balazs, J.A., Velásquez, J.D. 2016. Opinion Mining and Information Fusion: A survey. Information Fusion 27:95-110. doi:http://dx.doi.org/10.1016/j.inffus.2015.06.002

Blei, D.M., Ng, A.Y., Jordan, M.I. 2003. Latent Dirichl*et al*location. Journal of Machine Learning Research 3:993–1022

Brisebois, R, Abran, A., Nadembega, A. 2017. A Semantic Metadata Enrichment Software Ecosystem based on Metadata and Affinity Models. *International Journal of Information Technology and Computer Science* (IJITCS) 9 (8):1-13. doi:http://dx.doi.org/doi:10.5815/ijitcs.2017.08.01

Brisebois, R., Abran, A., Nadembega, A., N'techobo, P. 2017 A Semantic Metadata Enrichment Software Ecosystem based on Topic Metadata Enrichments. *International Journal of Data Mining & Knowledge Management Process* (IJDKP) 7 (3):1-23

Brisebois, R., Abran, A., Nadembega, A., N'techobo, P. 2017. A Semantic Metadata Enrichment Software Ecosystem based on Sentiment and Emotion Metadata Enrichments. *International Journal of Scientific Research in Science Engineering and Technology* (IJSRSET) 03 (02):625-641

Cambria, E., Gastaldo, P., Bisio, F., Zunino, R. 2015. An ELM-based model for affective analogical reasoning.

Neurocomputing 149, Part A:443-455. doi:http://dx.doi.org/10.1016/j.neucom.2014.01.064

Capilla, R., Bosch, J., Trinidad, P., Ruiz-Cortés, A., Hinchey, M. 2014. An overview of Dynamic Software Product Line architectures and techniques: Observations from research and industry. *Journal of Systems and Software* 91:3-23. doi:http://dx.doi.org/10.1016/j.jss.2013.12.038

Capilla, R., Jansen, A., Tang, A., Avgeriou, P., Babar, M.A.2016. 10 years of software architecture knowledge management: Practice and future. *Journal of Systems and Software* 116:191-205. doi:http://dx.doi.org/10.1016/j.jss.2015.08.054

Chen, P., Zhang, N.L., Liu, T., Poon, L.K.M., Chen, Z. 2016. Latent Tree Models for Hierarchical Topic Detection. arXiv preprint arXiv:160506650 [csCL]:1-44

Cho, H., Kim, S., Lee, J., Lee, J.S.2014. Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews. Knowledge-Based Systems 71:61-71. doi:http://dx.doi.org/10.1016/j.knosys.2014.06.001

Christensen, H.B., Hansen, K.M., Kyng, M., Manikas, K. 2014. Analysis and design of software ecosystem architectures – Towards the 4S telemedicine ecosystem. Information and Software Technology 56 (11):1476-1492. doi:http://dx.doi.org/10.1016/j.infsof.2014.05.002

Cigarrán, J., Castellanos, Á., García-Serrano, A. 2016 A step forward for Topic Detection in Twitter: An FCA-based approach. Expert Systems with Applications 57:21-36. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.011

Cotelo, J.M., Cruz, F.L., Enríquez, F., Troyano, J.A. 2016. Tweet categorization by combining content and structural knowledge. Information Fusion 31:54-64. doi:http://dx.doi.org/10.1016/j.inffus.2016.01.002

Dang, Q., Gao, F., Zhou, Y. 2016. Early detection method for emerging topics based on dynamic bayesian networks in micro-blogging networks. Expert Systems with Applications 57:285-295. doi:http://dx.doi.org/10.1016/j.eswa.2016.03.050

Dastidar, B.G., Banerjee, D., Sengupta, S. 2016. An Intelligent Survey of Personalized Information Retrieval using Web Scraper. *IJ Education and Management Engineering* 5:24-31. doi:10.5815/ijeme.2016.05.03

Demir, K.A.2015. Multi-View Software Architecture Design: Case Study of a Mission-Critical Defense System. *Computer and Information Science* 8 (4):12-31

Desmet, B., Hoste, V. 2013 Emotion detection in suicide notes. Expert Systems with Applications 40 (16):6351-6358. doi:http://dx.doi.org/10.1016/j.eswa.2013.05.050

Ferrara, E., De Meo, P., Fiumara, G., Baumgartner, R. 2014.Web data extraction, applications and techniques: A survey. Knowledge-Based Systems 70:301-323. doi:http://dx.doi.org/10.1016/j.knosys.2014.07.007

Gawer, A., Cusumano, M.A. 2014. Industry Platforms and Ecosystem Innovation. *Journal of Product Innovation Management* 31 (3):417-433. doi:http://dx.doi.org/ 10. 1111/jpim.12105

Gottlob, G., Koch, C., Pieris, A. 2017. Logic, Languages, and Rules for Web Data Extraction and Reasoning over Data. In: Drewes F, Martín-Vide C, Truthe B (eds) Language and Automata Theory and Applications: 11th International Conference, LATA 2017, Umeå, Sweden, March 6-9, 2017, Proceedings. Springer *International Publishing, Cham,* pp 27-47. doi:10.1007/978-3-319-53733-7_2

He, W., Xu, L.D. 2014. Integration of Distributed Enterprise Applications: A Survey. *IEEE Transactions on Industrial Informatics* 10 (1):35-42. doi:10.1109/TII.2012.2189221

Horcas, J.M., Pinto, M., Fuentes, L. 2016. An automatic process for weaving functional quality attributes using a software product line approach. *Journal of Systems and Software* 112:78-95. doi:http://dx.doi.org/10.1016/j.jss.2015.11.005

Kadam, V.B., K. Pakle, G. 2014. A Survey on HTML Structure Aware and Tree Based Web Data Scraping Technique. *International Journal of Computer Science and Information Technologies* 5 (2):1655-1658

Kapidakis, S., Houssos, N., Stamatis, K., Koutsourakis, P. 2015. Flexible metadata mapping using OAI-PMH. Paper presented at the Proceedings of the 8th ACM International Conference on PErvasive Technologies Related to Assistive Environments, Corfu, Greece,

Kiritchenko, S., Zhu, X., Mohammad, S.M. 2014 Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research* 50 (1):723-762. doi:http://dx.doi.org/10.1613/jair.4272

Kouzis-Loukas, D 2016. Learning Scrapy. Packt Publishing Ltd,

Kumar Roy, B., Chandra Biswas, S., Mukhopadhyay, P. 2017. Designing Metadata Harvesting Framework for OAI-based LIS Repositories: A Prototype. *International Journal of Information Science and Management* 15 (1):73-88

Manikas, K., Hansen, K.M. 2013. Software ecosystems – A systematic literature review. *Journal of Systems and Software* 86 (5):1294-1306. doi:http://dx.doi.org/10.1016/j.jss.2012.12.026

Mathew, A., Balakrishnan, H., Palani, S. 2015. Scrapple: a Flexible Framework to Develop Semi-Automatic Web Scrapers. *International Review on Computers and Software* 10 (5):475-480. doi:https://doi.org/10.15866/ irecos.v10i5.5864

Munezero, M.D., Montero, C.S., Sutinen, E., Pajunen, J. 2014. Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text. IEEE Transactions on Affective Computing 5 (2):101-111. doi:http://dx.doi.org/10.1109/TAFFC.2014.2317187

Neves, A.RdM, Carvalho, Á.M.G., Ralha, C.G. 2014. Agent-based architecture for context-aware and personalized event recommendation. Expert Systems with Applications 41 (2):563-573. doi:http://dx.doi.org/10.1016/j.eswa.2013.07.081

Olyai, A., Rezaei, R.2015. Analysis and Comparison of Software Product Line Frameworks. *Journal of Software* 10 (8):991-1001 doi:10.17706/jsw.10.8.991-1001

Patel, D., Thakkar, A.2015. A Survey of Unsupervised Techniques for Web Data Extraction. *International Journal of Computer Science & Communication* 6 (2):1-3

Patel, G.A., Madia, N. 2016. A Survey: Ontology Based Information Retrieval For Sentiment Analysis. *International Journal of Scientific Research in Science, Engineering and Technology* 2 (2):460-465

Quan, C., Ren, F. 2014 Unsupervised product feature extraction for feature-oriented opinion determination. Information Sciences 272:16-28. doi:http://dx.doi.org/10.1016/j.ins.2014.02.063

Ravi, K., Ravi, V. 2015. A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. Knowledge-Based Systems 89:14-46. doi:http://dx.doi.org/10.1016/j.knosys.2015.06.015

Serrano-Guerrero, J., Olivas, J.A., Romero, F.P., Herrera-Viedma, E.2015. Sentiment analysis: A review and comparative analysis of web services. Information Sciences 311:18-38. doi:http://dx.doi.org/10.1016/j.ins.2015.03.040

Singh, P.K., Sangwan, O.P., Singh, A.P., Pratap, A. 2015. A Framework for Assessing the Software Reusability using Fuzzy Logic Approach for Aspect Oriented Software. IJ *Information Technology and Computer Science* 7 (2):12-20. doi:10.5815/ijitcs.2015.02.02

Sufyan, D., Arjumand, M, Abdul Qayume, K., K. MP, Raza, Q.Ali, N. 2016. A Review - Web Scrapper Tool for Data Extraction. 2 (1):614-616

Teli, S. 2015. Metadata Harvesting From Selected Institutional Digital Repositories in India: A Model to Build a Central Repository. *International Journal of Innovative Research in Science, Engineering and Technology* 4 (4):1935-1942. doi:10.15680/IJIRSET.2015.0404018

Vilares, D., Alonso, M.A., GÓMez-RodrÍGuez, C. 2015. A syntactic approach for opinion mining on Spanish reviews. Natural Language Engineering 21 (1):139-163. doi:http://dx.doi.org/10.1017/S1351324913000181

Yadav, H.B., Yadav, D.K. 2015. A fuzzy logic based approach for phase-wise software defects prediction using software metrics. Information and Software Technology 63:44-57. doi:http://dx.doi.org/10.1016/j.infsof.2015.03.001

Zhang, C., Wang, H., Cao, L., Wang, W., Xu, F. 2016. A hybrid term–term relations analysis approach for topic detection. Knowledge-Based Systems 93:109-120. doi:http://dx.doi.org/10.1016/j.knosys.2015.11.006

****