


Article

A Semantic Focused Web Crawler Based on a Knowledge Representation Schema

Julio Hernandez ¹, Heidi M. Marin-Castro ^{2,*}  and Miguel Morales-Sandoval ¹

¹ Cinvestav Tamaulipas, Cd. Victoria, Tamps. 87130, Mexico; nhernandez@tamps.cinvestav.mx (J.H.); miguel.morales@cinvestav.mx (M.M.-S.)

² Cátedras CONACYT—Universidad Autónoma de Tamaulipas, Cd. Victoria, Tamps. 87000, Mexico

* Correspondence: heidy.marinc@gmail.com

Received: 4 April 2020; Accepted: 16 April 2020; Published: 31 May 2020



Abstract: The Web has become the main source of information in the digital world, expanding to heterogeneous domains and continuously growing. By means of a search engine, users can systematically search over the web for particular information based on a text query, on the basis of a domain-unaware web search tool that maintains real-time information. One type of web search tool is the semantic focused web crawler (SFWC); it exploits the semantics of the Web based on some ontology heuristics to determine which web pages belong to the domain defined by the query. An SFWC is highly dependent on the ontological resource, which is created by domain human experts. This work presents a novel SFWC based on a generic knowledge representation schema to model the crawler's domain, thus reducing the complexity and cost of constructing a more formal representation as the case when using ontologies. Furthermore, a similarity measure based on the combination of the inverse document frequency (IDF) metric, standard deviation, and the arithmetic mean is proposed for the SFWC. This measure filters web page contents in accordance with the domain of interest during the crawling task. A set of experiments were run over the domains of computer science, politics, and diabetes to validate and evaluate the proposed novel crawler. The quantitative (harvest ratio) and qualitative (Fleiss' kappa) evaluations demonstrate the suitability of the proposed SFWC to crawl the Web using a knowledge representation schema instead of a domain ontology.

Keywords: crawling; semantic focused web crawler; knowledge representation schema; web pages; similarity

1. Introduction

According to the website Live Stats [1], there are more than one billion of active websites on the World Wide Web (WWW). As a result, the increasing necessity of faster and reliable tools to effectively search and retrieve web pages from a particular domain has been gaining importance. One of the most popular tools to systematically collect web pages from the WWW are web crawlers. A web crawler is a system based on Uniform Resource Locator (URL) indexing to traverse the Web. URLs indexing provides a better service to web search engines and similar applications to retrieve resources from the web [2]. The web crawler searches for any URL reachable from the web page being retrieved by the search engine. Each URL found by the crawler is placed in a search queue to later be accessed by the search engine. The process repeats for each new URL retrieved from the queue. The stop criterion for URL searching varies; the most common is until reaching a threshold in the number of URLs retrieved from a seed or when reaching a level of depth.

The architecture of a web crawler is composed of three main components: (i) a URL frontier, (ii) the page downloader, and (iii) a repository. The Frontier stores the URLs that the web crawler

has to visit. The page downloader retrieves and parses the web pages from the URLs in the frontier. Finally, the downloaded web pages are stored in the repository component [3].

From the huge amount of resources in the web, most of them could be irrelevant to the domain of interest. This is why focused web crawlers (FWC) are better preferred to retrieve web pages. An FWC is based on techniques such as machine learning (classification) to identify relevant web pages, adding them to a local database [4]. An FWC (Figure 1) adds to the traditional crawler architecture a topic classifier module. This module is featured-based, modeling an input target domain to classify relevant web pages. If the web page is positively classified, its URLs are extracted and queued in the frontier module. In some FWC approaches [5–8], the classification module is based on document similarity metrics to filter related and non-related web pages to a given domain. However, these approaches do not take into account the expressiveness of web pages content, that is, they do not explore their semantic content or use that information in the filtering process.

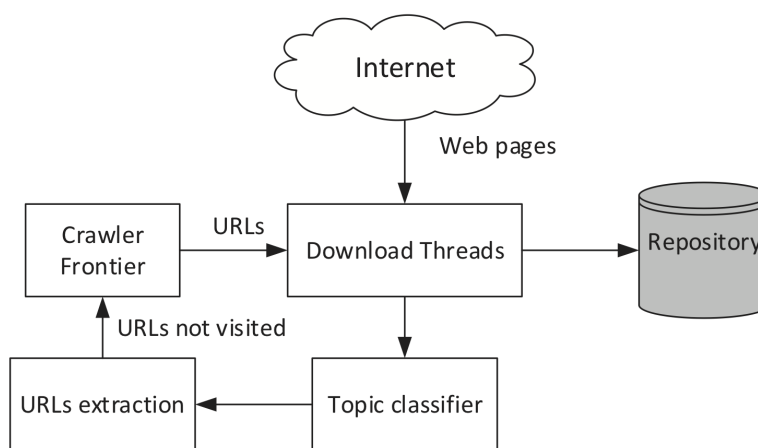


Figure 1. General focused web crawler architecture.

An FWC retrieves a set of topic-related web pages from a set of seed URLs. A seed URL is the starting point to iteratively extract URLs. That is, an FWC analyzes the content of seed URLs to determine the relevance of their content for a target domain. Such content analysis is based on techniques like ontology-based, machine learning, query expansion, among others [9]. Some approaches require an initial dataset to create a model (machine learning approaches [10]) or a set of keywords to produce specific domain queries (query expansion [11]).

The Semantic Web (SW), considered as an extension of today's Web, is based on a resource description framework (RDF) to express information in a well-defined meaning [12]. The SW arranges data as a logically linked data set instead of a traditional hyperlinked Web. An FWC that exploits the semantics of the Web content and uses some ontology heuristics is called Semantic Focused Web Crawler (SFWC). An ontology is a specification of a conceptualization, describing the concepts and relationships that can exist between domain's elements [13]. An SFWC determines the relevance of a web page to a user's query based on domain knowledge related to the search topic [14].

An SFWC performs two main tasks [15]: (i) content analysis and (ii) URL traversing (crawling). The content analysis task consists of determining if a web page is relevant or not for the topic given by the user's query. Algorithms such as PageRank [16], SiteRank [17], Visual-based page segmentation (VIPS) [18], and densometric segmentation [19] are well known web page content analyzers. The URL traversing task has as objective to define the order in which URLs are analyzed. Techniques like breadth-first, depth-first, and best-first are representative traversing strategies for this task [15].

In an SFWC, an ontology is commonly used to determine if a web page is related to the domain, comparing its text content with the ontology structure through similarity measures such as the cosine similarity [20] or the semantic relevance [12]. The use of a domain specific ontology helps to face

problems like heterogeneity, ubiquity, and ambiguity [21] since a domain ontology defines classes and their relationships, limiting its scope to predefined elements.

The main limitation of any SFWC is its dependency to the domain ontology being used, with particularly two main issues [22]: (i) an ontology is designed by domain experts, limiting their representation to the experts' understanding on the domain and (ii) data are dynamic and constantly evolving.

As an alternative to classic SFWC designs that use ontologies, this work presents a novel SFWC based on a generic knowledge representation schema (KRS) to model a target domain. The KRS analyzes the content of a document to identify and extract concepts, i.e., it maps the content of a document, from an input corpus, to an SW representation. The KRS, generated from each document, is stored in a knowledge base (KB) [23] to provide access to their content. The KRS is less expressive than a domain ontology (it does not define any rule or restriction over the data), but it is domain independent. Ontology-based approaches are structures whose concepts and relations are predefined by domain experts. Additionally, a similarity measure is proposed based on the inverse document frequency (IDF) measure and statistical measures such as arithmetic mean and standard deviation to compute the similarity between a web page content against the KRS. Our proposed SFWC is simple to build without the complexity and cost of constructing a more formal knowledge representation such as a domain ontology, but keeps the advantage of using SW technologies like RDFS (Resource Description Framework Schema).

In summary, the main contributions of this work are:

- A new SFWC based on a KRS,
- A generic KRS based on SW technologies to model any domain,
- A methodology to build a KRS from an input corpus,
- A similarity measure based on IDF and the statistical measures of arithmetic mean and standard deviation to determine the relevance of a web page for a given topic.

The proposed KRS builds a KB from an input corpus without an expert intervention, i.e., the KRS is based on content, representing entities as the most important element in a domain.

The rest of this paper is organized as follows. Section 2 discusses the related work. Section 3 presents the methodology for the construction of the SFWC and the similarity measure. Section 4 presents the results from the experiments. Finally, Section 5 concludes this work.

2. Related Work

This section presents relevant SFWC approaches proposed in the literature and the most recurrent metrics to measure the web page similarity in a given domain ontology.

SFWC approaches [22,24,25] exploit the expressiveness of an ontology to compute the similarity of a web page content against a domain ontology. Table 1 summarizes some ontology-based SFWC targeting different tasks, describing the measure used to determine the web page relevance. As it is shown, the cosine similarity is the most common measure used to determine the relevance of a web page against an ontology content.

SFWCs could be applied to different domains such as recommendation systems [25–27] or cybercrime [28], as Table 1 shows. In all cases, a specific domain ontology must define the most relevant elements and their relationship in the given domain. These approaches leave aside the semantic analysis of the source content, which could be exploited to better discrimination of web resources related to the domain. The proposed SFWC tries to alleviate the aforementioned situation, providing a semantic analysis to represent the relationship between content (words) and source (documents) through the KRS. The proposed KRS defines a set of classes with certain properties based on the SW standard RDFS. The KRS is a lightweight version of an ontology since it does not define complex elements like axioms or formal constraints but it is also based on SW technologies. The KRS depends on the input corpus to model a topic, i.e., the content information of the corpus is used to

generate the KRS. The schema provides an incremental feature, i.e., the KRS could be expanded with more specific domain documents since entities are independent between them but related by the source, e.g., all words from the same document are linked together.

Table 1. Representative ontology-based SFWCs.

Task	Description	Measure
Cloud service recommendation system [25,26]	A concept ontology-based recommendation system for retrieving cloud services.	Semantic relevance
Website models [29]	An ontology-supported website model to improve search engine results.	Cosine similarity
Web directory construction [20]	Based on a handmade ontology from WordNet to automatically construct a web directory.	Cosine similarity
User-based recommendation system [27]	A knowledge representation model built from a user interest database to select seeds URLs.	Concept similarity
Concept labeling [5,22]	An ontology-based classification model to label new concepts during the crawling process, integrating new concepts and relations to the ontology.	Cosine similarity, semantic relevance
Cybercrime [28]	Enhanced crime ontology using ant-miner focused crawler.	Significance

Traditional SFWCs are based on metrics like semantic relevance or cosine similarity to determine the relevance of a web page to a given domain. This kind of metric is used to measure the distance between two elements in an ontology. TF-IDF is a metric that has been used by different approaches to characterize a corpus and build a classification model [30–32]. Wang et al. [30] present a Naive Bayes classifier based on TF-IDF to extract the features of a web page content. Pesaranghader et al. [31] propose a new measure called Term Frequency-Information Content as an improvement of TF-IDF to crawl multi-term topics. A multi-term topic is a compound set of keywords that could not be eliminated to kept the meaning of the whole topic, e.g., web services. Peng et al. [32] present a partition algorithm to segment a web page into content blocks. TF-IDF was used also to measure the relevance of content blocks and to build a vector space-model [33] to retrieve topic and genre-related web pages. Kumar and Vig [34] proposed a Term-Frequency Inverse-Document Frequency Definition Semantic (TIDS). TIDS is a table of words associated with the sum of all TF-IDF values in the corpus. Hao et al. [35] proposed the combination of TF-IDF with LSI (Latent Semantic Indexing) to improve crawling results.

TF-IDF has been also used as a feature space to built structures in tasks like machine learning, multi-term topic, content block analysis, and table indexing to create complex models to determine the similarity between a document and a target domain. For example, a classification model based on TF-IDF requires a test and training set to generate a model, and the addition of more documents could lead to generate a new model based on a new test and training set and to compute TF-IDF values again. If a new document is added to the corpus, an FWC based on TF-IDF needs to compute again this value over all document's words in the corpus. In this work, we proposed the use of IDF as similarity measure since it provides the importance of a word in the corpus. The computation of IDF is faster in comparison with TF-IDF since it only needs to be computed for the words in the corpus and not for each word in a document in the corpus. The arithmetic mean and standard deviation are used to provided a dynamic threshold to define the similarity between a web page and the target domain.

3. Methodology

The proposed methodology for the KRS-based SFWC is divided into two general steps: (i) the KRS construction and (ii) the SFWC design. The following subsections explain each step in detail.

3.1. The KRS Construction

The SW provides a set of standards to describe data and their metadata. The resource description framework (RDF) is the SW standard to describe structured data. The basic element of RDF is known as triple. A triple is composed of a subject, object, and a predicate to define the relationship between them. A set of related triples are known as RDF graph. The SW also provides additional standards to

define more complex structures like ontologies, being RDFS and the Ontology Web Language (OWL) the standards for this purpose. These standards define the rules to build ontologies; however, RDFS is less expressive than OWL since it does not define restrictions or rules over the ontology.

The KRS (Figure 2) is a general and domain-free structure to describe the entities from a text source. In this work, a corpus is represented as a set of KRS stored in a KB. The goal of the KB is to provide the mechanisms to query the content of the KRS to measure the similarity between a web page content and the KRS.

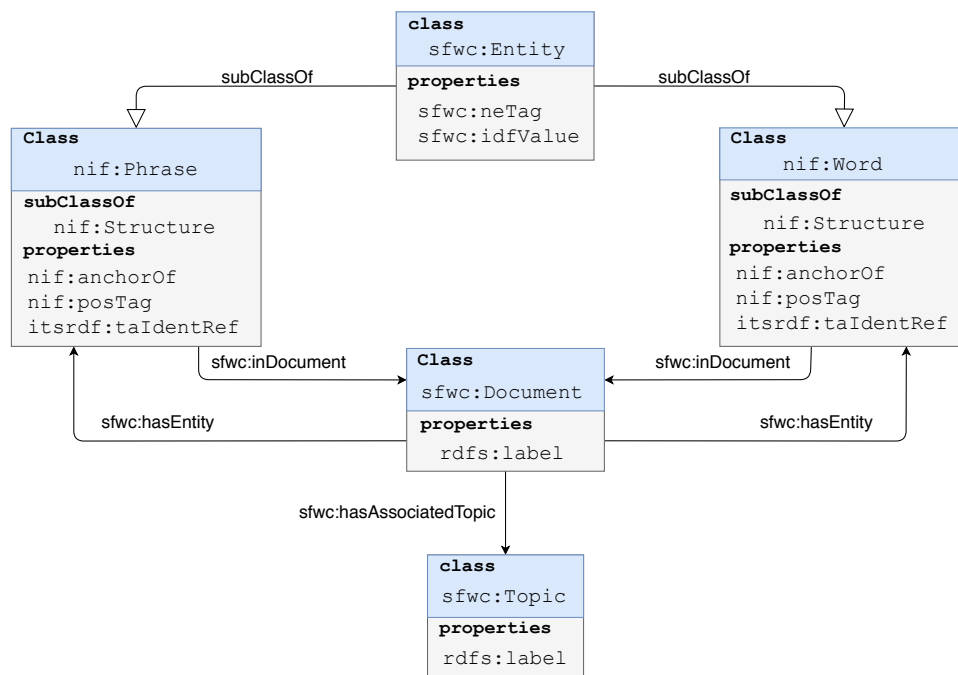


Figure 2. The Knowledge Representation Schema to describe the relationship between a noun and a corpus.

The KRS is based on RDF and RDFS to define topic entities and relationships. It is built considering the well known NIF (The NLP Interchange Format) [36] vocabulary which provides interoperability between language resources and annotations. In the KRS, a word is an instance of *sfwc:Entity*, representing a word as a phrase (*nif:Phrase*) or as a single word (*nif:Word*). Each word is described considering the following elements: (i) lemma word (*nif:anchorOf*), (ii) NE tag (*sfwc:neTag*), (iii) Part of Speech Tagging (PoS Tag) (*nif:posTag*), (iv) url (*itsrdf:taIdentRef*) and (v) IDF (*sfwc:idfValue*). These elements are used to determine if a new document is related to the target topic.

A document instance (*sfwc:Document*) is described only by the title of the source document, and it is related to the target topic (*sfwc:Topic*). The steps followed to populate the KRS are the following (Figure 3):

- A. Text processing
 - Preprocessing: The content of each document is processed for subsequent analysis. At this stage, stop words are removed and words are labeled with its corresponding PoS Tag and lemma.
 - Noun enrichment (NNE): The enrichment process assigns to each noun a PoS Tag, lemma, NE label, semantic annotation, and their IDF value. The PoS Tag and lemma were extracted in the previous step. The NE label and the semantic annotation are identified by Named Entity Recognition (NER) and Named Entity Linking (NEL) algorithms over the text. NER identifies NEs from an input text and NEL disambiguates those NEs to a KB to assign

a unique identifier (URL). The IDF measure assigns a weight to each noun in accordance with their importance in the corpus.

- B. Mapping process
 - RDF triple representation: The enriched nouns information is used to populate the KRS. A document is represented by a set of enriched nouns which are described by a set of RDF triples.
 - KRS generation: The KRS is generated from each document and stored in the KB.

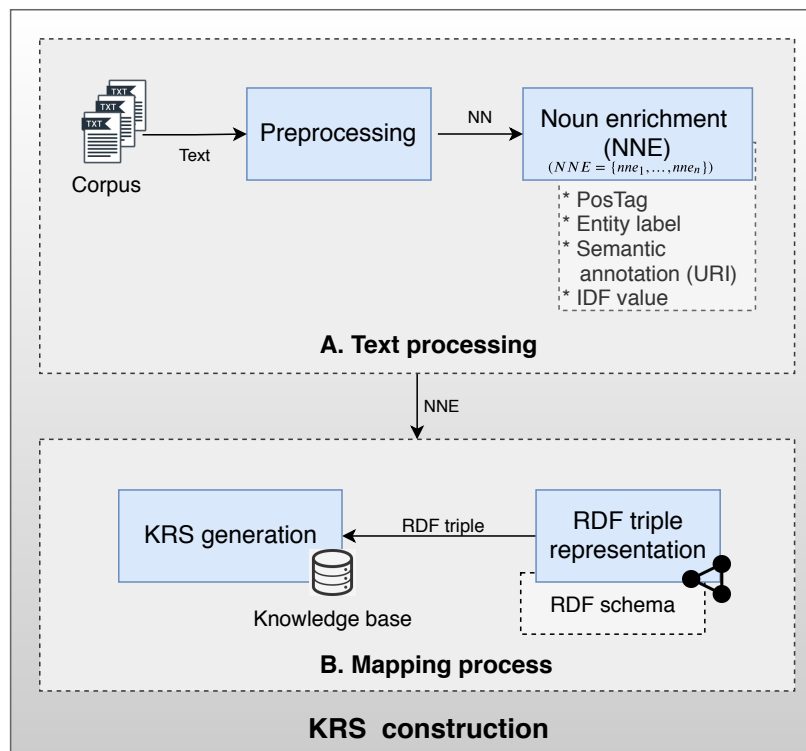


Figure 3. The KRS construction steps.

After the KRS is constructed, it is added to the topic classifier step of the SFWC process.

3.2. SFWC Design

The proposed SFWC (Figure 4) was inspired by the basic structure of an SFWC, whose main element is the topic classifier. The topic classifier determines if the content from a web page is related or not to the target topic or domain. Traditional approaches integrate a domain ontology in the topic classifier step. The domain ontology provides a predefined knowledge about the domain or topic. It describes the relationship between domain elements and could define some rules and restrictions. The KRS is an alternative to the use of domain ontologies, providing a simple schema to represent a topic specific content.

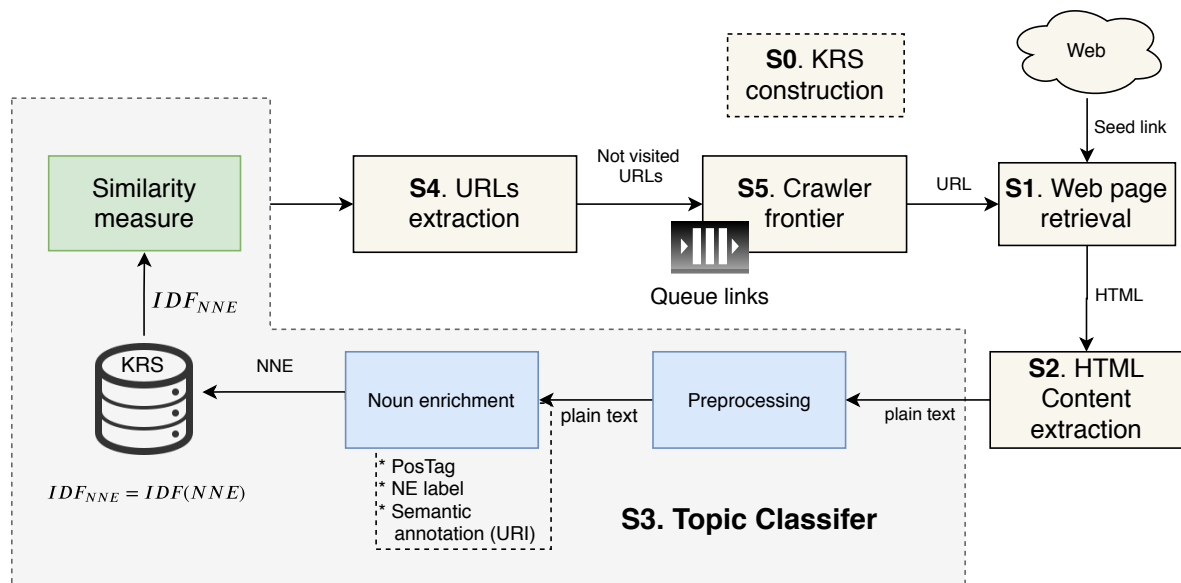


Figure 4. Overview of the proposed SFWC.

The proposed SFWC takes a seed web page as input to start the process and a queue of links to collect the URLs related with the target topic. In general, the proposed SFWC is divided in the following steps:

- S0. KRS construction: It represents a previous step in the SFWC process.
- S1. Web page retrieval: It downloads the web page to be locally analyzed.
- S2. HTML content extraction: It extracts the web page content.
- S3. Topic classifier: It processes the web page content and analyzed it to determine if it is similar or not with the KRS.
- S4. URLs extraction: It extracts the URLs (enclosed in <a> tags) from the original web page content.
- S5. Crawler frontier: It collects and stores the extracted URLs in a queue of links. The crawler frontier sends the next URL to the web page retrieval step.

The first and second steps (S1 and S2) are focused on getting the content from a seed web page. One of the core components of the proposed SFWC is the topic classifier, which is constructed in the third step. Like the domain ontology-based approaches, the topic classifier performs a content analysis of web pages to determine their similarity with the KRS. The topic classifier begins with the text preprocessing and the NNE tasks. These tasks have the same purpose as in the KRS construction. The IDF value for each enriched noun is computed from the KRS using SPARQL queries. SPARQL is the SW standard to query RDF triples from a KB or an RDF graph. In this work, a query retrieves the number of documents containing the noun extracted from the web page content. The retrieved results must match the noun anchor text and they must be described by the same URL.

The similarity measure, described in the next section, calculates the arithmetic mean with respect to the extracted IDF from each enriched noun. In this work, it is established that a web page content is similar to the KRS if the arithmetic mean is within a threshold.

The last two steps (S4 and S5) extract the corresponding URLs and store them in a queue links. The process from S1 to S5 is repeated until the queue is empty or the process reaches a predefined number of iterations.

3.3. Similarity Measure

The proposed SFWC compares web page's content against the KRS. The goal is to determine if a web page is closely related with a target domain considering the input corpus. The system takes into

account the enriched nouns from a source to compare their content against the KRS. Our proposed approach uses the IDF and the statistics measures of arithmetic mean and standard deviation to calculate the similarity between the web page content and the KRS.

Some SFWC proposals [32,34,35] are based on the use of TF-IDF as similarity measure. TF is a statistical measure to define the weight or importance of each word from a document. IDF is a statistical measure to define the importance of each word with respect to the corpus. The combination of these measures define the weight of each word with respect to the document and the corpus.

The main issue with TF-IDF is that a noun can be weighted with different TF-IDF value in accordance with their corresponding document, i.e., a noun from different documents will have different TF-IDF values. To create a unique value per word, a method [34] was proposed to calculate the average of the TF-IDF value for each word; however, this value must be updated if the corpus increases their number. The proposed similarity measure is based on IDF (Equation (1)) since it defines a unique value for each noun with respect to a corpus, and it is easily updated if the number of documents increases:

$$IDF(t, C) = \log(N/n) \quad (1)$$

where t is a term (word) in a document, C is a corpus, N is the number of documents in the corpus, and n is the number of documents containing the target word. The IDF metric tends to be high for uncommon words and low for very common words. However, there is no specification about the ideal IDF value to determine the relevance of a word in the corpus. Equations (2)–(4) define respectively the arithmetic mean, standard deviation, and the similarity measure used in this work:

$$\mu_C = \sum IDF(t_i, C) \quad (2)$$

where t_i is an enriched noun whose URI value is not empty. The arithmetic mean is calculated over enriched nouns whose description is linked to a KB of the SW, e.g., Wikidata, DBpedia, etc.

$$\sigma = \sqrt{\frac{\sum (idf(t_i) - \mu)^2}{N}} \quad (3)$$

$$sim_{doc} = \mu_{doc} \therefore \mu_C - \sigma \leq \mu_{doc} \leq \mu_C + \sigma \quad (4)$$

μ is the arithmetic mean of IDF values in the corpus (μ_C) or in a document (μ_{doc}), and σ represents the standard deviation calculated from IDF values. In this work, μ and σ define the threshold used to determine the similarity between a web page and the KB. This threshold is calculated as: $\mu(IDF) \pm \sigma$.

The similarity measure was inspired in normal distribution where the threshold tries to represent frequent words and uncommon words, that is, we suppose that relevant words are in the range of $\mu \pm \sigma$, i.e., the similarity measure selects the most representative words described in the KRS. The calculated threshold is used as a reference to determine whether the content of a web page is related to the KRS or not.

4. Implementation and Experiments

This section presents the implementation of the KRS and the SFWC and the evaluation of the proposed SFWC.

4.1. Implementation

4.1.1. KRS Implementation

The proposed method was evaluated over three topics from Wikipedia: (i) computer science, (ii) politics, and (iii) diabetes.

The implementation of the KRS construction is divided into three steps: (i) corpus gathering, (ii) text processing, and (iii) mapping process.

In the corpus gathering step, the documents for each topic from Wikipedia online encyclopedia are collected. However, it could be used any other source rather than Wikipedia pages—for example, a specific set of domain related documents or a specific corpus from repositories such as kaggle (<https://www.kaggle.com/>) or the UCI repository (<https://archive.ics.uci.edu/ml/index.php>). The Wikipedia encyclopedia is an open collaboration project, and it is the general reference work on the World Wide Web [37–39]. It tends to have better structured, well-formed, grammatical and meaningful, natural languages' sentences compared to raw web data [40]. Table 2 shows the number of pages extracted for each topic, the depth of the extraction system, and the restriction set. The depth extraction refers to the number of subtopics extracted for each topic and the restriction is the filtering rule to select the Wikipedia pages.

Table 2. The number of web pages extracted for the three different categories.

Category	URLs	Depth	Restriction
Computer science	1151	1	Pages describing
Diabetes	202	1	persons are
Politics	1717	1	not considered
Total	3070		

After building the corpus for each topic, the next step is to generate the corresponding KRS.

Text Processing

Figure 5 shows the KRS generation. Each document is analyzed to extract enriched nouns through NLP and SW technologies. The Stanford core NLP tool splits a document's content into sentences and extracts information like PoS Tags, lemmas, indexes, and NEs. Additionally, each sentence is analyzed with DBpedia spotlight to look for entities linked to the DBpedia KB, retrieving the corresponding URL. The tasks involved in this process are:

- Sentence splitting: The content of a document is divided into sentences, applying splitting rules and pattern recognition to identify the end of a sentence.
- Lemmatization: The root of each word is identified, e.g., the lemma of the verb *producing* is *produce*.
- PoS Tagging: A label is assigned to each token in a sentence, indicating their part of speech in the sentence, e.g., NN (noun), advj (adjective), PRP (personal pronoun), etc. PoS Tag are labels from the Penn treebank (a popular set of part of speech tags used in the literature).
- NER: NEs identification. The result is a single-token tagged with the corresponding NE (person, location, organization, etc.).
- NEL: Entities are defined in an SW KB. From a set of candidates' words, each word is a query against the target KB to retrieve a list of possible matching entities. After a ranking process, the most relevant entity is selected and their URL is returned and associated with the corresponding word.

According with Figure 5, the first task identifies and extracts enriched nouns from the corpus and store them in the NOSQL DB MongoDB. Then, the relevance of an enriched noun is computed based on the statistical measure IDF.

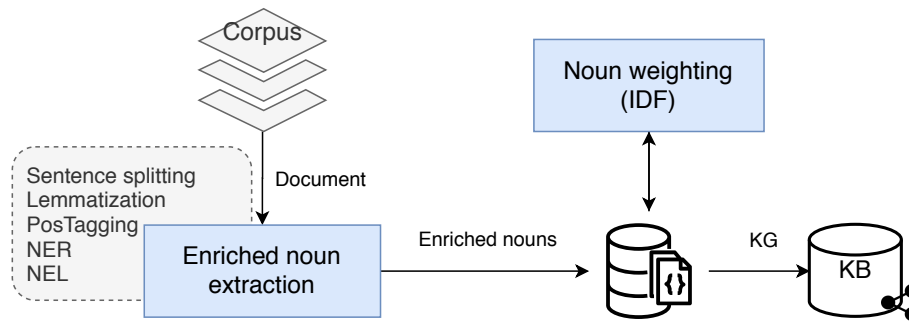


Figure 5. KRS generation steps, from a given document’s corpus.

Mapping Process

The KRS is produced from MongoDB, where enriched nouns are mapped to the KRS as RDF triples and stored in a KB.

The KB provided the basic functionality of querying over RDF triples. It is set up in a SPARQL endpoint to query their content and retrieve the data needed to compute the similarity between a web page content and the KB.

An example of the KRS is shown in Figure 6. The figure shows the Atkins_diet resource of type document (Basal_rate, 15-Anhydroglucitol and Artificial_pancreas are also of type document), associated with the topic of diabetes. The document contains five NEs linked to DBpedia KB (astrup, approach, appetite, analysis, Atkins).

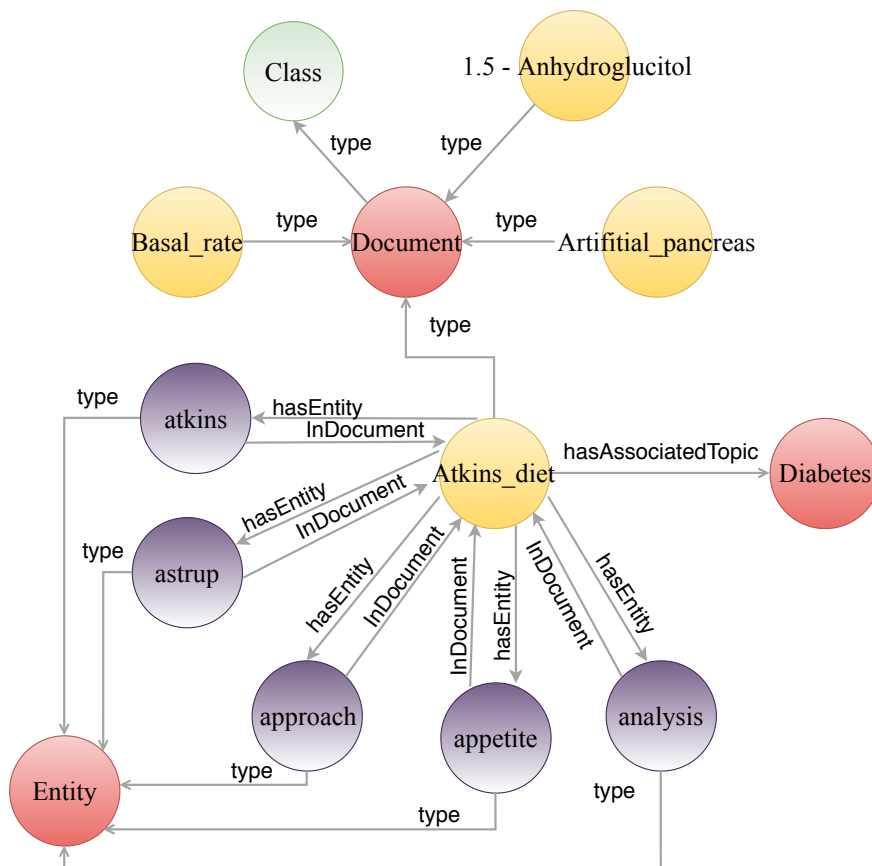


Figure 6. An excerpt of the diabetes KB.

4.1.2. The SFWC Implementation

The implementation of the proposed SFWC is explained in the following paragraphs.

Web Page Retrieval

The first step retrieves a web page from an input seed URL or from a queue of URLs. This module implements two methods to select the set of seed URLs as input for the proposed SFWC: (i) querying a search engine about a topic and (ii) randomly selecting a set of seed URLs from the input corpus. In the first case, the Google search API was used to query and retrieve seed URLs. The API allows for setting up a personalized Google search engine to query. For the experiments, the first five page results from Google were collected (50 URLs). In the second case, the same number of URLs (50 URLs) was randomly selected from the input corpus as in the first case.

HTML Content Extraction

The second step was implemented with the Java library Jsoup. The library contains functions to extract the content of predefined HTML tags, e.g., the $\langle p \rangle$ tag defines a paragraph. Jsoup is used to retrieve the text enclosed by this tag.

Topic Classifier

The Stanford Core NLP tool was used to analyze the web pages content, defining the PoS Tag, lemma, and entity label. DBpedia Spotlight was used to define the semantic annotation for each noun. The enriched nouns are used to compute the similarity of the web page content against the KRS. In this case, if the web page content is similar to the KRS content, the web page is stored in a repository of related web pages.

URL Extraction

This module implements a breadth-first approach to extract and add URLs to the crawler frontier. Figure 7 illustrates the breadth-first approach (part A) and how they are stored in the queue of URLs (part B).

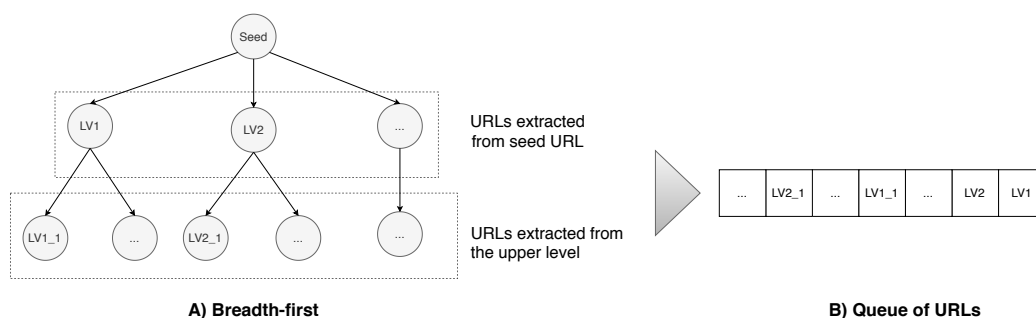


Figure 7. Breadth-first URL extraction process (A) and queue URLs (B) after adding the extracted URLs. LV denotes the level of each node.

Crawler Frontier

The crawler frontier was implemented as a queue of URLs, arranged in accordance with the breadth-first algorithm.

4.2. Results and Evaluation

The experiments were executed in an iMac with a 3 GHz Intel Core i5 processor (Victoria, Tamps, Mexico), 16 GB of RAM and macOS Mojave as an operating system. The implemented application was developed in Java 8.

The experiments were conducted over three different corpuses, built from the Wikipedia categories of computer science, politics, and diabetes. A KRS was constructed to represent the content of each corpus. The relevance of a web page content in a given topic was computed using a similarity measure based on the statistical measure IDF and a threshold defined by the arithmetic mean and the standard deviation. Table 3 shows the statistics of the three Wikipedia categories. The number of Wikipedia pages retrieved for each category corresponds to the first level of the category hierarchy. For example, the root level of computer science category contains 19 subcategories and 51 Wikipedia pages. For each subcategory, the corresponding Wikipedia pages are extracted, resulting in 1151 documents (second column in Table 3). The third column presents the total number of enriched nouns extracted and the average enriched nouns per document. The fourth column shows the total number of enriched nouns with a URL associated with a KB of the SW and the average value per Wikipedia page.

Table 3. Category information per topic and enriched noun extraction statistics.

Category	Number of Pages	Total Enriched Nouns (Average per Document)	Total Enriched Nouns Associated with a KB (Average per Document)
Computer science	1151	289,950 (251.91)	24,993 (21.71)
Diabetes	202	83,723 (414.47)	14,470 (71.63)
Politics	1717	793,137 (461.66)	80,024 (46.58)
TOTAL	3070	1,166,810 (380.07)	119,487 (38.92)

The results from experiments were analyzed qualitatively and quantitatively. The first one is focused on the number of downloaded web pages related to a topic. The second one is focused in the quality of the results from the quantitative experiments.

4.2.1. Qualitative Results

The proposed SFWC was evaluated over two sets of seed URLs from a different source: (i) seed URLs retrieved from the Google search engine and (ii) seed URLs selected from the built corpus (Wikipedia category). Tables 4 and 5 show the results per topic after processing both sets of seed URLs. The first column corresponds to the topic. The second column is associated with the number of seed URLs retrieved from the Google search engine and Wikipedia. In the case of the search engine, it was queried with the topic name, e.g., the query string “computer science” was used to retrieve the web pages related with the topic of computer science. For the case of Wikipedia, the set of seed URLs was randomly selected for each category from the built corpus, e.g., 50 Wikipedia pages were randomly selected from the politics corpus. The last three columns show a summary of the processed seed URLs: (i) crawled, (ii) not crawled, and (iii) not processed Wikipedia pages. The *seed URLs crawled* column defines the number of seed URLs whose content was similar to the corresponding topic after computing the similarity measure, i.e., the similarity measure result was in the threshold. The *seed URLs not crawled* column defines the seed URLs whose content was not similar to the corresponding topic, i.e., the similarity measure result was not in the threshold. The last column (*seed URLs not processed*) defines the number of seed URLs that was not processed because an error occurred, e.g., the seed URL returns the HTTP 400 error code (Bad Request Error). That means that the request sent to the website server was incorrect or corrupted and the server couldn’t understand it. The results from Google’s seed URLs (Table 4) got the lowest number of seed URLs crawled in comparison with the Wikipedia’s seed URLs results (Table 5) in which all topic seed URLs crawled are above 50%. Additionally, the Google’s

seed URLs were prone to errors, being the most recurrent the HTTP 400 error code (bad URL request). In contrast, Wikipedia's seed URLs were not prone to these kinds of errors.

Table 4. Google page results statistics.

Topic	Seed URLs (Google)	Seed URLs Crawled	Seed URLs Not Crawled	Seed URLs Not Processed
Computer science	50	11 (22%)	32 (64%)	7 (14%)
Diabetes	50	9 (18%)	30 (60%)	11 (22%)
Politics	50	22 (44%)	18 (36%)	10 (20%)
Total	150	42 (28%)	80 (53.33%)	28 (18.67%)

Table 5. Corpus Wikipedia pages statistics.

Topic	Seed URLs (Wikipedia)	Seed URLs Crawled	Seed URLs Not Crawled	Seed URLs Not Processed
Computer science	50	26 (52%)	24 (48%)	0 (0%)
Diabetes	50	31 (62%)	19 (38%)	0 (0%)
Politics	50	39 (78%)	10 (22%)	1 (2%)
Total	150	96 (64%)	53 (36%)	1 (0.6%)

The URLs crawled from the seed URLs were not restricted or limited to be from the same domain name as the seed URL (e.g., <http://en.wikipedia.org>), that is, the URLs added to the queue could be from any other domain different from the seed URL. Tables 6 and 7 show the crawling results for each topic. The first column defines the topic. The second column defines the number of seed URLs crawled. The number of seed URLs processed is in correspondence with the third column (seed URL crawled) from Tables 4 and 5. The *domain names crawled* column defines the number of different domain names crawled, e.g., the diabetes topic contains the lowest number of seed URLs crawled, but it is the second topic with the highest number of domain names crawled, which means that the seed URLs in the diabetes topic are connected to many other domains. The *Web pages analyzed* column defines the total number of web pages analyzed by the proposed SFWC. The columns *Accepted*, *Rejected*, and *Error* distribute the number of the web pages crawled into those whose content is related with the corresponding topic, not related with the corresponding topic and the web pages that could not be processed due to an error (e.g., HTTP 400 error) because the URL was an image (e.g., <http://example.com/image.jpg>) or a PDF file. In accordance with the results, the crawled Wikipedia's seed URLs obtained the highest number of web pages accepted (the web page content is related with the corresponding topic), but it also contains the highest number of errors, produced because the URLs contain an image (jpg files) instead of text.

In accordance with the results from Tables 6 and 7, seed URLs from Wikipedia obtained the best results in comparison with those obtained with the seed URLs from Google. Seed URLs from Google contain information from several domain names. The seed URLs from Wikipedia contain almost only URLs to other Wikipedia pages. Additionally, the seed URLs from Wikipedia have the same web page structure and format. The seed URLs from Google do not share the same structure, and the content could be in different formats. In this sense, URLs crawled from Google were less accepted because the

domain names are heterogeneous, and the content could drastically change in format and structure from one URL to another.

Table 6. Google results for crawled seed URLs.

Topic	Seed URLs	Domain Names Crawled	Web Pages Analyzed	Accepted	Rejected	Error
Computer science	11	104	874	86 (9.84%)	765 (87.52%)	23 (2.63%)
Diabetes	9	135	957	265 (27.69%)	605 (63.22%)	87 (9.09%)
Politics	22	182	1893	754 (39.83%)	1113 (58.79%)	26 (1.37%)
Total	42	421	3724	1105 (29.67%)	2483 (66.67%)	136 (3.65%)

Table 7. Wikipedia results for crawled seed URLs.

Topic	Seed URLs	Domain Names Crawled	Web Pages Analyzed	Accepted	Rejected	Error
Computer science	26	51	2624	1101 (41.96%)	1488 (56.71%)	35 (1.33%)
Diebetes	31	55	3119	1413 (45.30%)	1670 (53.54%)	36 (1.15%)
Politics	39	8	3910	2781 (71.12%)	1097 (28.06%)	32 (0.82%)
Total	96	114	9653	5295 (54.85%)	4255 (44.08%)	103 (1.06%)

Evaluation

The evaluation is based on the Harvest Ratio [31,32,41] (HR) measure shown in Equation (5). According to Samarawickrama and Jayaratne [42], the HR is the primary metric to evaluate a crawler performance. The HR measure the rate at which relevant web pages are acquired and irrelevant web pages are filtered off from the crawling process:

$$\text{Harvest Ratio} = \frac{||R_p||}{||T_p||} \quad (5)$$

where R_p corresponds to those web pages accepted by the system and evaluated as correct and T_p corresponds to the total accepted web pages downloaded by the SFWC, evaluated as correct or incorrect.

Similarity Measure

The similarity measure is based on the statistical measure IDF computed over the enriched nouns with an URL associated with a KB from the SW. The similarity measure of a web page content against the KRS is calculated as follows:

1. The arithmetic mean (μ) and standard deviation (σ) for the KB is computed over all enriched nouns whose URL value is not empty.
2. For every new web page content, enriched nouns are extracted.
3. The IDF value for the new web page is calculated over all enriched nouns whose URL value is not empty.

- If the arithmetic mean of the web page content is between $\mu \pm \sigma$, the web page is accepted. Table 8 defines the threshold range for each topic.

Equation (6) shows the process to compute the IDF value for the enriched noun “*algol*”, where N is the total number of documents in the computer science topic and n is the number of documents containing the enriched noun “*algol*”. The computed IDF value is 6.36 which is added to the IDF values calculated from the remaining enriched nouns from the web page content:

$$\begin{aligned}
 IDF(t, C) &= \log(N/n) \\
 &= \log(1151/14) \\
 &= \log(82.21) \\
 &= 6.36
 \end{aligned}
 \tag{6}$$

To illustrate this process, Listing 1 shows the query used to retrieve the number of documents (?total) containing the word “*algol*” from the computer science topic. The returned value corresponds to the divisor (n) in the IDF equation. The dividend (N) value is retrieved by the query shown in Listing 2, returning the total number of documents in the KB (the subjects whose type is *sfwc:Document*).

Listing 1: SPARQL query to retrieve the number of documents containing the word “*algol*” from the KRS.

```

@PREFIX sfwc: <http://sfwcrawler.com/core#>
@PREFIX nif: <http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#>
SELECT COUNT (DISTINCT ?doc) as ?total WHERE {
?s a sfwc:Entity .
?s nif:anchorOf "algol" .
?s sfwc:inDocument ?doc .
}
    
```

Listing 2: The SPARQL query to the KRS to retrieve the number of documents.

```

@PREFIX sfwc: <http://sfwcrawler.com/core#>
SELECT COUNT (DISTINCT ?doc) as ?total WHERE {
?doc a sfwc:Document .
}
    
```

Table 8. Threshold range by topic.

Topic	Arithmetic Mean (μ)	Standard Deviation (σ)	Threshold Range ($\mu \pm \sigma$)
Computer science	5.01	1.56	[3.45,6.57]
Diabetes	3.19	1.54	[1.65,4.73]
Politics	4.84	1.54	[3.30,6.38]

The evaluation was conducted by four human raters and performed over a stratified random sample of the crawled web pages. This kind of sample was selected to maintain consistency in the results since a human rater evaluates the results from each topic. The first step of the stratified random sample consists of calculating the sample size from the whole data (see Equation (7)):

$$n = \frac{N\sigma^2Z^2}{(N - 1)\epsilon^2 + \sigma^2Z^2}
 \tag{7}$$

where n is the sample size, N is the size of the corpus, σ is the standard deviation, Z is the confidence value, and ϵ is the sample error rate. The second step consists of calculating the sample size for the accepted and rejected web pages (see Equation (8)):

$$n_i = n * \frac{N_i}{N} \tag{8}$$

where n_i corresponds to the sample size of accepted or rejected web pages, N_i is the total web pages for accepted or rejected, and N is the size of the corpus. Table 9 shows the sample values for accepted and rejected web pages for Google (G) and Wikipedia (W). The sample, for Google and Wikipedia, was randomly selected.

Table 9. Sample size calculation for accepted and rejected web pages. G is for Google data and W is for Wikipedia data.

Topic	Total Examples		Sample Size (N)		Sample Size for Accepted (n_1)		Sample Size for Rejected (n_1)	
	G	W	G	W	G	W	G	W
Computer science	879	2589	303	335	93	142	210	192
Diabetes	957	3083	284	342	29	157	255	185
Politics	1893	3878	374	350	151	251	223	99

Tables 10 and 11 show the HR results for each rater and the summary per topic for the seed URLs from Google and Wikipedia, respectively.

Table 10. Harvest rate values for Google seed URLs per rater. R1 is for rater 1, R2 is for rater 2, etc.

	R1	R2	R3	R4	Average
Computer science	70/93 (75.27%)	74/93 (79.57%)	65/93 (69.89%)	65/93 (69.89%)	68.5/93 (73.66%)
Diabetes	26/29 (89.66%)	23/29 (79.31%)	26/29 (89.66%)	28/29 (96.55%)	25.75/29 (88.79%)
Politics	110/151 (72.85%)	106/151 (70.20%)	93/151 (61.59%)	114/151 (75.50%)	105.75/151 (70.03%)

Table 11. Harvest Rate values for Wikipedia seed URLs per rater. R1 is for rater 1, R2 is for rater 2, etc.

	R1	R2	R3	R4	Average
Computer science	119/142 (83.80%)	110/142 (77.46%)	111/142 (78.17%)	106/142 (74.65%)	111.5/142 (78.52%)
Diabetes	127/157 (77.71%)	132/157 (84.08%)	124/157 (78.98%)	134/157 (85.35%)	128/157 (81.53%)
Politics	217/251 (86.45%)	240/251 (95.62%)	225/251 (89.64%)	206/251 (82.07%)	222/251 (88.45%)

According to the results from the Tables 10 and 11, the proposed SFWC was consistent with the results for the seed URLs from Google and Wikipedia. These results demonstrate that the KRS and the similarity measure selects the most relevant concepts for each topic. The KRS describes the nouns from the input corpus and the arithmetic mean and the standard deviation establish a threshold to determine which nouns are the most representative for the topic. The similarity measure defines if web page content is related to the given topic or not if the result is between the predefined threshold. The combination of KRS and the similarity measure help to select the most related web pages.

The best results were obtained with the diabetes topic which is a more specific topic than computer science and politics. The average value for the computer science and diabetes topics is closed, whereas, for political topics, there is an important difference for Google and Wikipedia.

The computer science and politics topics contain several subtopics, e.g., the root level of the Wikipedia category of computer science contains 18 subcategories, the category of politics contains 38 subcategories, and the category of diabetes contains 10 subcategories. The corpus for each topic was built only with the first level of the Wikipedia category, leaving aside a significant number of Wikipedia pages, e.g., Table 12 shows the number of Wikipedia pages for the first five levels.

The average results obtained by computer science and politics are promising since it does not contain the whole Wikipedia pages from their corresponding categories. The diabetes category is a more specialized category, containing specific terms of the topic and, as can be seen in Table 12, the number of Wikipedia pages does not exponentially increase level by level. The average results obtained with the diabetes topic are better than those obtained with the remaining categories. In the particular case of the diabetes topic for Google results, the number of seed URLs crawled was 9 and the total number of web pages analyzed was 957, resulting in 265 accepted web pages. These numbers are lower in comparison with the seed URLs crawled from Wikipedia; however, the average percentage is quite similar, even when the number of accepted web pages are too different.

Table 12. The number of Wikipedia pages by category level.

Category	Levels				
	0	1	2	3	4
Computer science	44	1151	6487	26,730	79,845
Diabetes	145	202	336	349	357
Politics	63	1717	17,346	86,260	291,615

4.2.2. Qualitative Results

The qualitative evaluation was conducted using the Fleiss’ kappa measure, shown in Equation (9). The Fleiss’ kappa is an extension of Cohen’s kappa which is a measure of the agreement between two raters, where agreement due to chance is factored out. This case, the number of raters can be more than two. As for Cohen’s kappa, no weighting is used and the categories are considered to be unsorted:

$$\kappa = \frac{\bar{p} - \bar{p}_e}{1 - \bar{p}_e} \tag{9}$$

where \bar{p} defines the actual observed agreement and \bar{p}_e represents chance agreement. The factor $\bar{p} - \bar{p}_e$ represents the degree of agreement actually achieved above chance and the factor $1 - \bar{p}_e$ represents the degree of agreement that is attainable above chance. κ takes the value of 1 if the raters are in complete agreement.

The results obtained by the human raters in the quantitative evaluation are analyzed with the Fleiss’ kappa measure. Table 13 shows the results for each topic and for each seed URLs source (Google and Wikipedia). Table 14 shows the interpretation agreement between raters. According with these values, Wikipedia’s seed URLs obtained a substantial agreement between the human raters; meanwhile, Google’s seed URLs obtained a moderate agreement (computer science and politics) and substantial agreement for diabetes. The diabetes corpus was consistent in the qualitative evaluation in both cases (Wikipedia and Google).

Table 13. κ value for seed URLs from Wikipedia and Google for the three corpuses.

Corpus	Wikipedia	Google
	$(\kappa \text{ value})$	
Computer science	0.68	0.53
Diabetes	0.65	0.62
Politics	0.63	0.57

Table 14. κ interpretation table of agreement.

κ	Interpretation
<0	Poor agreement
0.01–0.20	Slight agreement
0.21–0.40	Fair agreement
0.41–0.60	Moderate agreement
0.61–0.80	Substantial agreement
0.81–1.00	Almost perfect agreement

Discussion

In accordance with the results from Tables 10 and 11, the average HR for each topic is above 70%, that is, the accepted or downloaded web pages are relevant to the corresponding topic. The computer science and politics topics got an average HR under the 80% since both topics are broader than the diabetes topic, i.e., the computer science and politics topics contain several Wikipedia pages as is pointed out in Table 12, e.g., the fourth level for computer science contains 79,845 Wikipedia pages and the diabetes topic contains 357 Wikipedia pages at the same level.

The SFWC relies on the proposed KRS to describe the content of a corpus from any topic. In the evaluation, the corpus size does not determine the quality of the crawling results. The quality was determined by the content of the corpus, and the selection of the most representative enriched nouns for each corpus in the KRS. For example, the diabetes corpus size is 202, and it was the topic with the best results in the quantitative and qualitative analysis. However, the computer science and politics topics could improve the results if the corpus increases their size since the number of Wikipedia pages per level has a significance difference, as it is shown in Table 12.

5. Conclusions and Future Work

This work presented a novel semantic focused web crawler (SFWC) based on a knowledge representation schema (KRS), as an alternative to traditional SFWCs that use domain ontologies designed by human experts. The KRS has the feature to model any domain, is less complex, less formal, and easier to build than an ontology. The KRS describes the most relevant elements in the domain and can be automatically constructed through a semantic analysis of an input corpus. Even with a relatively low number of input web pages used to construct the corpus, as it was the case with the Wikipedia pages in this work, the average results are promising as the SFWC was able to filter relevant web pages with a score above 70%, endorsed by human raters.

As part of the mechanisms for the SFWC to filter web pages, a new metric was used, by combining the IDF and statistical measures. The achieved results demonstrated the high capacity (above 69%) of the proposed SFWC to filter relevant web page content based on a quantitative and qualitative evaluation, being more effective with specialized topics such as the diabetes topic whose vocabulary terms have a close relation among them and thus the content of web pages associated with that domain.

The quantitative and qualitative results demonstrate that the proposed SFWC reaches a substantial agreement between the human raters, obtaining better results (with a score above 80%) with the diabetes topic, which is more specific than the politics and computer science topics (score above 70%).

As future work, alternative approaches will be explored to select the input web page corpus for the KRS construction, that is, to select the most relevant topic's documents as well as to extend the evaluation to broader topics.

Author Contributions: Conceptualization, J.H. and H.M.M.-C.; Data curation, J.H.; Formal analysis, H.M.M.-C.; Investigation, J.H. and H.M.M.-C.; Methodology, J.H., H.M.M.-C. and M.M.-S.; Project administration, H.M.M.-C. and M.M.-S.; Software, J.H.; Supervision, H.M.M.-C. and M.M.-S.; Validation, J.H., H.M.M.-C. and M.M.-S.; Visualization, H.M.M.-C. and M.M.-S.; Writing—original draft, J.H.; Writing—review & editing, J.H., H.M.M.-C. and M.M.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by “Fondo Sectorial de Investigación para la Educación”, CB SEP-CONACyT Mexico, project number 281565.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Internet Live Stats—Internet Usage and Social Media Statistics. 2016. Available online: <https://www.internetlivestats.com/> (accessed on 19 April 2020).
- Lu, H.; Zhan, D.; Zhou, L.; He, D. An Improved Focused Crawler: Using Web Page Classification and Link Priority Evaluation. *Math. Prob. Eng.* **2016**, *2016*, 6406901. [CrossRef]
- Udapure, T.V.; Kale, R.D.; Dharmik, R.C. Study of Web Crawler and its Different Types. *IOSR J. Comput. Eng.* **2014**, *16*, 1–5. [CrossRef]
- Kumar, M.; Vig, R. Learnable Focused Meta Crawling Through Web. *Procedia Technol.* **2012**, *6*, 606–611. [CrossRef]
- Gaur, R.K.; Sharma, D. Focused crawling with ontology using semi-automatic tagging for relevancy. In Proceedings of the 2014 Seventh International Conference on Contemporary Computing (IC3), Noida, India, 7–9 August 2014; pp. 501–506.
- Du, Y.; Liu, W.; Lv, X.; Peng, G. An Improved Focused Crawler Based on Semantic Similarity Vector Space Model. *Appl. Soft Comput.* **2015**, *36*, 392–407. [CrossRef]
- Kumar, J. *Apache Solr Search Patterns*; Packt Publishing Ltd.: Birmingham, UK, 2015.
- Salah, T.; Tiun, S. Focused crawling of online business Web pages using latent semantic indexing approach. *ARPN J. Eng. Appl. Sci.* **2016**, *11*, 9229–9234.
- Kumar, M.; Bhatia, R.K.; Rattan, D. A survey of Web crawlers for information retrieval. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2017**, *7*. doi:10.1002/widm.1218. [CrossRef]
- Priyatam, P.N.; Vaddepally, S.R.; Varma, V. Domain specific search in indian languages. In Proceedings of the first ACM Workshop on Information and Knowledge Management for Developing Regions, Maui, HI, USA, 2 November 2012; pp. 23–29.
- Altingovde, I.S.; Ozcan, R.; Cetintas, S.; Yilmaz, H.; Ulusoy, O. An Automatic Approach to Construct Domain-Specific Web Portals. In Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, Lisbon, Portugal, 6–7 November 2007; Association for Computing Machinery: New York, NY, USA, 2007; pp. 849–852. doi:10.1145/1321440.1321558. [CrossRef]
- Bedi, P.; Thukral, A.; Banati, H.; Behl, A.; Mendiratta, V. A Multi-Threaded Semantic Focused Crawler. *J. Comput. Sci. Technol.* **2012**, *27*, 1233–1242. doi:10.1007/s11390-012-1299-8. [CrossRef]
- Gruber, T.R. A translation approach to portable ontology specifications. *Knowl. Acquis.* **1993**, *5*, 199–221. [CrossRef]
- Batzios, A.; Dimou, C.; Symeonidis, A.L.; Mitkas, P.A. BioCrawler: An Intelligent Crawler for the Semantic Web. *Expert Syst. Appl.* **2008**, *35*, 524–530. doi:10.1016/j.eswa.2007.07.054. [CrossRef]
- Yu, Y.B.; Huang, S.L.; Tashi, N.; Zhang, H.; Lei, F.; Wu, L.Y. A Survey about Algorithms Utilized by Focused Web Crawler. *J. Electron. Sci. Technol.* **2018**, *16*, 129. doi:10.11989/JEST.1674-862X.70116018. [CrossRef]
- Page, L.; Brin, S.; Motwani, R.; Winograd, T. The PageRank citation ranking: Bringing order to the Web. In Proceedings of the 7th International World Wide Web Conference, Brisbane, Australia, 14–18 April 1998; pp. 161–172.
- Wu, J.; Aberer, K. Using SiteRank for Decentralized Computation of Web Document Ranking. In *Adaptive Hypermedia and Adaptive Web-Based Systems*; De Bra, P.M.E., Nejd, W., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; pp. 265–274.

18. Cai, D.; Yu, S.; Wen, J.R.; Ma, W.Y. *VIPS: A Vision-based Page Segmentation Algorithm*; Technical Report MSR-TR-2003-79; Microsoft: Redmond, WA, USA, 2003.
19. Kohlschütter, C.; Nejdil, W. A Densitometric Approach to Web Page Segmentation. In Proceedings of the 17th ACM Conference on Information and Knowledge Management, Napa Valley, CA, USA, 26–30 October 2008; pp. 1173–1182. doi:10.1145/1458082.1458237. [[CrossRef](#)]
20. Khalilian, M.; Abolhassani, H.; Alijamaat, A.; Boroujeni, F.Z. PCI: Plants Classification Identification Classification of Web Pages for Constructing Plants Web Directory. In Proceedings of the 2009 Sixth International Conference on Information Technology: New Generations, Las Vegas, NV, USA, 27–29 April 2009; pp. 1373–1377. doi:10.1109/ITNG.2009.6. [[CrossRef](#)]
21. Patel, R.; Bhatt, P. A Survey on Semantic Focused Web Crawler for Information Discovery Using Data Mining Technique. *Int. J. Innov. Res. Sci. Technol.* **2014**, *1*, 168–170.
22. Hassan, T.; Cruz, C.; Bertaux, A. Ontology-based Approach for Unsupervised and Adaptive Focused Crawling. In Proceedings of the International Workshop on Semantic Big Data, Chicago, IL, USA, 19 May 2017; ACM: New York, NY, USA, 2017; pp. 2:1–2:6.
23. Krótkiewicz, M.; Wojtkiewicz, K.; Jodłowiec, M. Towards Semantic Knowledge Base Definition. In *Biomedical Engineering and Neuroscience*; Hunek, W.P., Paszkiel, S., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 218–239.
24. Khalilian, M.; Zamani Boroujeni, F. Improving Performance in Constructing specific Web Directory using Focused Crawler: An Experiment on Botany Domain. In *Advanced Techniques in Computing Sciences and Software Engineering*; Springer: Dordrecht, The Netherlands, 2010; pp. 461–466. doi:10.1007/978-90-481-3660-5_79. [[CrossRef](#)]
25. Boukadi, K.; Rekik, M.; Rekik, M.; Ben-Abdallah, H. FC4CD: A new SOA-based Focused Crawler for Cloud service Discovery. *Computing* **2018**, *100*, 1081–1107. doi:10.1007/s00607-018-0600-2. [[CrossRef](#)]
26. Ben Djemaa, R.; Nabli, H.; Amous Ben Amor, I. Enhanced semantic similarity measure based on two-level retrieval model. *Concurr. Comput. Pract. Exp.* **2019**, *31*, e5135, doi:10.1002/cpe.5135. [[CrossRef](#)]
27. Du, Y.; Hai, Y.; Xie, C.; Wang, X. An approach for selecting seed URLs of focused crawler based on user-interest ontology. *Appl. Soft Comput.* **2014**, *14*, 663–676. [[CrossRef](#)]
28. Hosseinkhani, J.; Taherdoost, H.; Keikhaee, S. ANTON Framework Based on Semantic Focused Crawler to Support Web Crime Mining Using SVM. *Ann. Data Sci.* **2019**, 1–14. [[CrossRef](#)]
29. Yang, S.Y. A Focused Crawler with Ontology-Supported Website Models for Information Agents. In *International Conference on Grid and Pervasive Computing*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 522–532. doi:10.1007/978-3-642-13067-0_54. [[CrossRef](#)]
30. Wang, W.; Chen, X.; Zou, Y.; Wang, H.; Dai, Z. A Focused Crawler Based on Naive Bayes Classifier. In Proceedings of the Third International Symposium on Intelligent Information Technology and Security Informatics, IITSI 2010, Jingtangshan, China, 2–4 April 2010; pp. 517–521. doi:10.1109/IITSI.2010.30. [[CrossRef](#)]
31. Pesaranghader, A.; Pesaranghader, A.; Mustapha, N.; Sharef, N.M. Improving multi-term topics focused crawling by introducing term Frequency-Information Content (TF-IC) measure. In Proceedings of the 2013 International Conference on Research and Innovation in Information Systems (ICRIIS), Kuala Lumpur, Malaysia, 27–28 November 2013; pp. 102–106. doi:10.1109/ICRIIS.2013.6716693. [[CrossRef](#)]
32. Peng, T.; Zhang, C.; Zuo, W. Tunneling enhanced by web page content block partition for focused crawling. *Concurr. Comput. Pract. Exp.* **2008**, *20*, 61–74. doi:10.1002/cpe.1211. [[CrossRef](#)]
33. Pappas, N.; Katsimpras, G.; Stamatatos, E. An Agent-Based Focused Crawling Framework for Topic- and Genre-Related Web Document Discovery. In Proceedings of the 2012 IEEE 24th International Conference on Tools with Artificial Intelligence, Athens, Greece, 7–9 November 2012; Volume 1, pp. 508–515. doi:10.1109/ICTAI.2012.75. [[CrossRef](#)]
34. Kumar, M.; Vig, R. Term-Frequency Inverse-Document Frequency Definition Semantic (TIDS) Based Focused Web Crawler. In *Global Trends in Information Systems and Software Applications*; Krishna, P.V., Babu, M.R., Ariwa, E., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 31–36.
35. Hao, H.; Mu, C.; Yin, X.; Li, S.; Wang, Z. An improved topic relevance algorithm for focused crawling. In Proceedings of the 2011 IEEE International Conference on Systems, Man, and Cybernetics, Anchorage, AK, USA, 9–12 October 2011; pp. 850–855. doi:10.1109/ICSMC.2011.6083759. [[CrossRef](#)]

36. Hellmann, S.; Lehmann, J.; Auer, S.; Brümmer, M. Integrating NLP Using Linked Data. In *The Semantic Web—ISWC 2013*; Alani, H., Kagal, L., Fokoue, A., Groth, P., Biemann, C., Parreira, J.X., Aroyo, L., Noy, N., Welty, C., Janowicz, K., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 98–113.
37. Lerner, J.; Lomi, A. The Third Man: hierarchy formation in Wikipedia. *Appl. Netw. Sci.* **2017**, *2*, 24. doi:10.1007/s41109-017-0043-2. [[CrossRef](#)] [[PubMed](#)]
38. Schrage, F.; Heist, N.; Paulheim, H. Extracting Literal Assertions for DBpedia from Wikipedia Abstracts. In *Semantic Systems. The Power of AI and Knowledge Graphs*; Acosta, M., Cudré-Mauroux, P., Maleshkova, M., Pellegrini, T., Sack, H., Sure-Vetter, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 288–294.
39. Wu, I.C.; Lin, Y.S.; Liu, C.H. An Exploratory Study of Navigating Wikipedia Semantically: Model and Application. In *Online Communities and Social Computing*; Ozok, A.A., Zaphiris, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 140–149.
40. Yano, T.; Kang, M. *Taking advantage of Wikipedia in Natural Language Processing*; Carnegie Mellon University, Pittsburgh, PA, USA, 2016.
41. Altingovde, I.S.; Ulusoy, O. Exploiting interclass rules for focused crawling. *IEEE Intell. Syst.* **2004**, *19*, 66–73. doi:10.1109/MIS.2004.62. [[CrossRef](#)]
42. Samarawickrama, S.; Jayaratne, L. Automatic text classification and focused crawling. In Proceedings of the 2011 Sixth International Conference on Digital Information Management, Melbourne, Australia, 26–28 September 2011; pp. 143–148. doi:10.1109/ICDIM.2011.6093329. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).