Using Concept Maps and Fuzzy Set Information Retrieval Model To Dynamically Personalize RSS Feeds

Heba M. Ismail,

American University of Sharjah, College of Engineering, Sharjah, UAE

Summary
Knowledge has always been at the heart of work and action of people. The mount of knowledge is growing and types of knowledge are becoming more diverse. The World Wide Web is becoming the dominant source of knowledge. With the speed pace of modern life, acquiring up-to-date knowledge as it becomes available on the web that exactly meets personal needs can define oneself competitive advantage. Therefore, it's becoming a pressing requirement. Web 2.0 has opened new opportunities for knowledge management with knowledge generation tools such as blogs, wikis, forums, really simple syndicate (RSS), social tagging and others. This paper presents a novel method for instantly updating web users with pieces of knowledge tailored to their personal knowledge interests retrieved from RSS feeds by combining the logic of fuzzy set information retrieval model with the semantics and structure of concept maps. The proposed Fuzzy Set IR model is based on custom fuzzy dictionary specific to the user's concept map. This in turn is different from any other information retrieval method which all work based on general thesaurus or indexes based on millions of users. Experiments show that the proposed method can achieve precision levels ranging between 80% and 100%.

Key words:
Just-In-Time Knowledge Management, Concept Map, Fuzzy Set Information Retrieval Model, Custom Fuzzy Dictionary, RSS Feed.

1. Introduction

The World Wide Web has revolutionized the way people access knowledge. However, knowledge on the Web is largely disorganized, scattered and untraceable. Substantial time is spent filtering out relevant knowledge that exactly meets user's interest. Search engines such as Google and Yahoo made it much easier to capture specific knowledge on the web. Yet, search engines works with limited number of keywords and generates results based on the web behavior of millions of users. Many attempts tried to address the problem of the limited number of keywords in the user's query by building interface systems that restructure users' query or by embedding semantics in the query by means of mind maps, concept maps and others [2], [3], [4], [5]. However, the root cause of the problem is still not addressed. Search engines like Google are based on algorithms that generate results ranked according to millions of users' behavior [13]. Therefore, search results may not closely match the specific interests of a single user. In addition, users need to be continuously updated on newly created knowledge on the web. Just-In-Time knowledge is becoming more critical to web users. Really Simple Syndicate (RSS) technology is employed to keep users updated on new web contents as they become available [6]. However, RSS feeds are unstructured and are far from being specific to a single user's actual knowledge interests even with the use of categories and groups [17], [20]. As a result a method that takes care of a single user's knowledge interests and provides Just-In-time knowledge relevant to these interests is required.

2. Objectives and Proposed Methodology

This paper addresses the following question: how to instantly update users on newly created knowledge that is specific to their knowledge interests as it becomes available on the web? Current search engines and smart RSS readers suffer from at least one of the following problems:

(i) Results are retrieved based on keywords rather than concepts.
(ii) Search engines allow limited number of keywords. (e.g. Google's input string is limited to 2048 bytes and 10 individual words) [9].
(iii) Results are ranked based on millions of user's rather than single user.
(iv) Users fail sometimes in defining their knowledge interests using limited number of keywords.
(v) Retrieval of knowledge is based on exact match of keywords rather than partial match of meaning.

This paper builds upon outcomes of prior research on using concept mapping to define the user's interest in Just-In-Time knowledge searching. RSS feeds are used to provide instant updates of newly created knowledge on the web, concept maps are proposed to represent user's knowledge interest where both the semantic and the structure of the concept map define the interest of the user, and Fuzzy Set Information Retrieval model (Fuzzy Set IR) [1] is used as an information retrieval logic that measures partial similarity and exact match between concepts in the concept map and terms in the RSS feed. The proposed

Manuscript received February 5, 2014
Manuscript revised February 20, 2014

Fuzzy Set IR model is based on custom fuzzy dictionary specific to the user's concept map. This in turn is different from any other information retrieval method which all work based on general thesaurus or indexes based on millions of users. A new method that combines the structure and semantic of the concept map with the sophisticated logic of fuzzy set IR model is proposed to solve the research question.

3. Information Retrieval – IR

The process of searching for relevant information from documents' corpora has always been a big concern for people in all fields. As an academic field of study, information retrieval (IR) can be defined as: '*finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)* [21:1].' The idea of using computers to search for relevant pieces of information was popularized in the article *"As We May Think"* by *Vannevar Bush* in 1945 [22]. The main components of any IR method are the user query, the matching algorithm, i.e. processor, document collection and user interface as shown in Fig.1 [23].

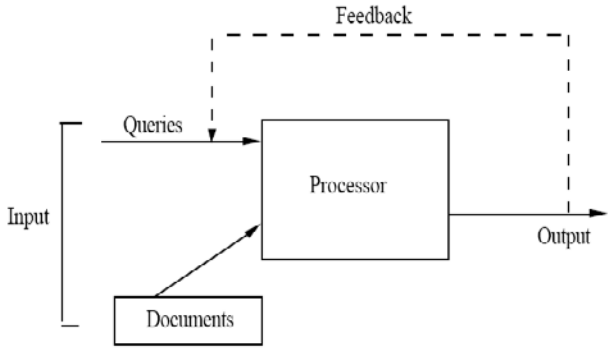

Fig. 1 Information Retrieval Process [23]

To assess the effectiveness of an IR method, two statistics are commonly used [21]:

(i) Precision: What fraction of the returned results is relevant to the information need? It's calculated as in Eq. (1).

Precision = # (relevant Items Retrieved) / #(Retrieved Items) (1)

(ii) Recall: What fraction of the relevant documents in the collection is returned by the method? It's calculated as in Eq. (2).

Recall = # (relevant Items Retrieved) / #(Relevant Items) (2)

Different IR methodologies were developed and proved different levels of effectiveness in different applications. Most common algorithms and the root of other methodologies are Vector Space, and Boolean Retrieval. In the latter, only documents that are true for the query are returned. Therefore, query is viewed as a set of keywords with connecting logical operators such as *AND*, *OR* and *Not*. Documents are viewed as set of keywords as well [1]. In Boolean retrieval, we deal with the exact match of keywords. Vector space model (VSM) computes a measure of similarity by defining vectors for each document in the collection and for the query. In VSM term's weights are used to indicate different importance level of different terms [1] [21]. Vector Space Model overcomes some problems inherent in the basic Boolean Retrieval Algorithm by assigning weights to the terms in the query and in the documents. This in turn allows for ranking different keywords differently. However, weights assigned in the VSM are not so formal and to some extent can be considered intuitive. Furthermore, terms are considered independent and no clear associations between terms are defined. Here comes the need for an information retrieval algorithm that has a formal method of weighting and considers the associations between terms.

3.1. Fuzzy Set Theory

Conventional set or crisp set [24] is any collection of objects which can be treated as a whole. Cantor described a set by its members, such that an item from a given universe is either a member or not. On the other side, a *Fuzzy Set* [25] contains items that belong to the set at different grades of membership. Following Zadeh, many sets have more than an Either-Or criterion for membership. Take for example the set of Young people. A one-year old baby will clearly be a member of the set, and a 100 years old person will not be a member of this set, but what about people at the age of 20, 30, or 40 years? Another example is a weather report regarding high temperatures, strong winds, or nice days. Zadeh proposed a *Grade of Membership*, such that the transition from membership to non-membership is gradual rather than abrupt. An item's grade of membership is normally a real number between 0 and 1, often denoted by the Greek letter μ. The higher the number, the higher the membership is. Zadeh regards Cantor's set as a special case where elements have full membership, μ = 1. He nevertheless called Cantor's sets non-Fuzzy; today the term crisp set is used, which avoids that little dilemma. Zadeh does not give a formal basis for how to determine the grade of membership. The grade of membership is a precise, but subjective measure that depends on the context. A membership function $fA: C \rightarrow$

[0,1] is used to indicate the degree of membership of each element in the set.

3.2. Fuzzy Set Information Retrieval Model

As explained in section 3.1 fuzzy set theory relies on two main principles: (1) sets are not well defined, and (2) elements belong to the fuzzy set at different levels of membership. Language sentences and documents are typical examples of fuzzy sets. A Fuzzy Set IR model is adopted to determine the degree of membership between every keyword in a sentence and a fuzzy set that contains different words each belongs to the set at some degree of membership. The degrees of similarity or membership, also referred as the correlation factors among words, are given by a function which assigns a value in the range [0, 1] to any two words. Hence, if two sentences contain many terms that belong to the same fuzzy sets at high degree of membership then the two sentences are similar. There are several methods to define the correlation factors among different words.

3.3. Defining the Fuzzy Association and Membership Function

In 1991, [14] adopted a fuzzy set IR model to determine whether a keyword in a sentence belongs to a fuzzy set that contains words with different levels of similarities amongst themselves. [14] called the fuzzy set a *keyword-connection-matrix* and defined it as a kind of thesaurus (i.e. dictionary) which described relations between keywords by assigning similarity grades restricted to the interval [0,1]. Later in 2005, [15] used the same keyword-connection-matrix proposed by [14] to detect similar HTML documents. Using the keyword-connection-matrix [15] compared every keyword *k* in sentence *i* with every keyword *w* in document *d* and calculated a word-sentence similarity using the following fuzzy association:

$$\mu_{k,d} = 1 \_ \prod_{w \in d} (1 - C_{kw}) \quad (3)$$

where C_{kw} is the fuzzy relationship between *k* and *w*. The average of all μ-value is calculated to find the overall similarity between sentence *i* and document *d* as follows:

$$\text{Sim}(i,d) = \frac{\mu_{k1,d} + \mu_{k2,d} + \dots + \mu_{n1,d}}{n} \quad (4)$$

, where n is the number of keywords *k* in sentence *i*. relying on the concept of co-occurrence alone to determine similarity or relationship between words is sometimes inaccurate. For example if we consider a document that talks about computer architecture the keyword connection factor between "computer" and "architecture" would be high. In another document that talks about how architects use computers in their design again the keyword connection factor will be high as both words "computer" and "architecture" would appear in the same document quite often. And both factors calculated for each document might be the same which is not accurate and doesn't reflect the actual relationship between the two keywords. In 2008 [18] and [19] adopted the same fuzzy set IR model adopted by [14] and [15] to cluster similar RSS feeds. However, [18] and [19] didn't rely on the keyword-connection-matrix only to define the fuzzy membership or the correlation factor, instead they used three different types of correlation factors: (1) word Connection, (2) word co-occurrence, and (3) distance. Later [26] proved that the correlation factors in the distance matrix provided the most accurate results amongst all the three matrices with accuracy rate of 94% compared to 47% for the keyword-connection and 52% for the Co-occurrence. That is because distance correlation factors accounts for frequency and co-occurrence at the same time.

3.4. Advantages of Using Fuzzy Set IR Model

Fuzzy set information retrieval model provides solutions for problems inherent in other IR methods such as VSM and Boolean models as explained earlier. It proved effectiveness in dealing with ill-defined information. Furthermore, fuzzy set IR model proved to work well on partially related semantics and similar sentences along with exact match.

4. Concept Mapping

Concept mapping [10] was developed in educational settings by Joseph Novak, in an effort to design better teaching and learning method. Concept maps are graphical tools for organizing and representing knowledge. They include concepts, usually enclosed in circles or boxes of some type, and relationships between concepts indicated by a connecting line linking two concepts. In concept maps, the concepts are represented in a hierarchical fashion with the most general concepts at the top of the map and the less general concepts arranged hierarchically below. The hierarchical structure for a particular domain of knowledge also depends on the context in which that knowledge is being applied. Therefore, it is best to construct concept maps with reference to some particular question we seek to answer, which is called a focus question. For example Fig.2 shows a concept map that addresses the question "What is a Plant?"

4.1. Why Use Concept Maps in Information Retrieval?

As explained earlier in this paper limited number of allowed keywords in search engines or other types of knowledge extractors on the web challenge the user to express his/her interest in the shortest possible sentence. Users many times fail to select good keywords that describe their interest or may even overlook some terms that can distinguish their interests from irrelevant domains of knowledge. Furthermore, users are not given the chance to personally assign different levels of importance to different keywords, rather, keywords weights are allocated based on algorithms that consider the behavior of millions of users on the web. Given this problem with keywords, previous research attempted to provide tools that help users better express their knowledge interests for more effective retrieval of relevant knowledge. First attempts were in the field of restructuring the user's query in order to eliminate the problem that many people face in selecting good query terms. In 1999 Yukio [2] proposed a "Supplying Keywords System" that divided the user's query into sub-queries (i.e. terms making up the query) in order to enhance results returned by a search engine. However, restructuring the user's query in all its flavors work on a bad query in the first place. It can be seen as a corrective rather than a protective solution. Other research attempted to start with the user by guiding the user in defining good queries from the beginning. Since concept maps and mind maps proved to be good tools for representing knowledge in educational context, [3], [4] and [5] proposed the use of a graph to express knowledge interest of specific users. Using a graph, users can completely and thoroughly define their knowledge interest without having to squeeze it into ten-word query or less. In addition, graphs have a hieratical representation which indicates the importance of different concepts as conceived by specific users.

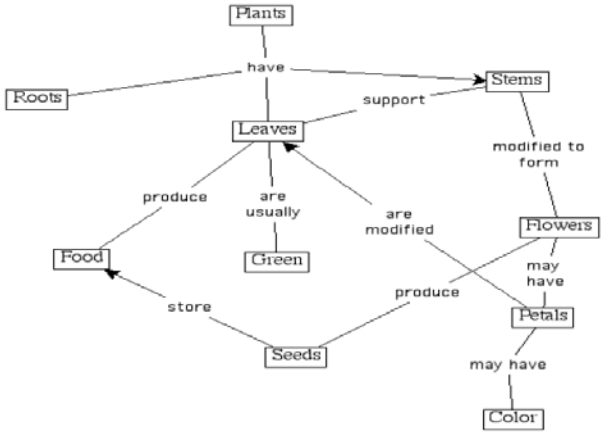

Fig. 2 What Is a Plant Concept Map [10]

4.2. Concept Maps Topology

A fundamental assumption shared by almost all knowledge organization theories dictates that knowledge can be modeled in terms of set of concepts and their relationships. Concept mapping is a method for making such concepts and relationships within individual's brain explicit. Starting from this idea research started to consider the layout and topology of the concept map in defining the user interest instead of relying only on the concepts and their relationships. [4] Proposed that the closer two concepts in the graph the closer their relationship will be and hence the more similar their search results should be. [11] And [12] studied further concept map's structure influences, considered incoming and outgoing connections, and proposed 3 models that help assigning structural or topological weights to every concept in the graph. As defined by [11] and [12] these models are Connectivity Root-Distance (CRD), Path Counter (PC) and Hub Authority and Root-Distance Model (HARD). In this paper we are considering these models to analyze the graph topology.

4.2.1. Connectivity Root-Distance Model (CRD)

The connectivity root-distance model is based on two observations. First, higher connectivity are considered more important. Second, the root concept, typically located at the top of a map, tends to be the most inclusive concept to the topic of the map. This suggests that concepts closer to the root are more important. The CRD model determines closeness to the root by counting the number of direct links between the map's root concept and a given concept. If concept k has o outgoing and i incoming connections to other concepts and is d steps distant from the root concept of the map, then the weight assigned to k by the CRD model is $(k) = \big(\alpha \,.\, o(k) + \beta \,.\, i\,(k)\big).(1/(d(k)+1))^{\,1/\gamma 2}$. The model fixed parameters α, β, γ determine influence of the incoming connections, outgoing connections, and distance to the root concept. The formula implies that the higher a concept's connectivity and the shorter its distance to the root concept, the larger its weight and therefore relevance to the topic of the map.

4.2.2. Path Counter Model (PC)

The Path Counter Model, like the CRD model, reflects the expectation that concepts participating in more propositions will tend to be more important to the topic of a map. However, instead of considering only a concept node's immediate connectivity the PC model considers

indirect relationships as well. It counts all possible paths, starting from the root concept, that contains the concept in question and end directly or indirectly on the concept under examination. Formally, to determine the weigh *W(k)* of a concept in *k* a map, assume that *n* is the number of paths that lead to *k* from the root concept, then the weight is computed as *W (k) = n.*

4.2.3. Hub Authority and Root-Distance Model (HARD)

While CRD performs a local analysis, only taking immediate neighbors into account, HARD performs a global analysis on the influences of the concepts on each other. Its analysis centers on three different types of concepts that may be found in any concept map:

(i) *Authorities* are concepts that have multiple incoming connections from hub nodes.
(ii) *Hubs* are concepts that have multiple outgoing connections to authority nodes.
(iii) *Upper nodes* include the root concept and concepts closest to the root concept.

To determine a node's role as a hub or authority, [11] adapted Kleinberg's algorithm [13] for analyzing hyperlinked graphs to concept maps. The adapted algorithm associates each concept with three weights between 0 and 1, each reflecting the concept's role as a hub, authority, or upper node. A given concept may simultaneously have properties of all three. In the HARD model, the three weights of a selected concept k are combined into a single weight as follows: $W(k) = \alpha . h(k) + \beta . a(k) + \gamma . u(k)$. In this formula h, a, u are the corresponding authority, hub, and upper node weights of a concept in a map and α, β, γ are the model fixed parameters.

5. Combining Concept Maps With Fuzzy Set IR Model

The proposed method is made up of number of processes as illustrated in Fig.3. There are three main processes:

(i) The process of generating the fuzzy set of words (i.e. the fuzzy dictionary) using distance correlation factor.
(ii) The process of analyzing the structure of the concept map, and calculating concept's weights.
(iii) The process of retrieving knowledge from RSS feeds using the fuzzy set and the concepts' weights.

The proposed processes were verified in a JAVA application that applies all the steps starting from creating the fuzzy dictionary and moving through defining specific RSS feeds and updating the user on relevant knowledge. Implementation is discussed later in this paper.

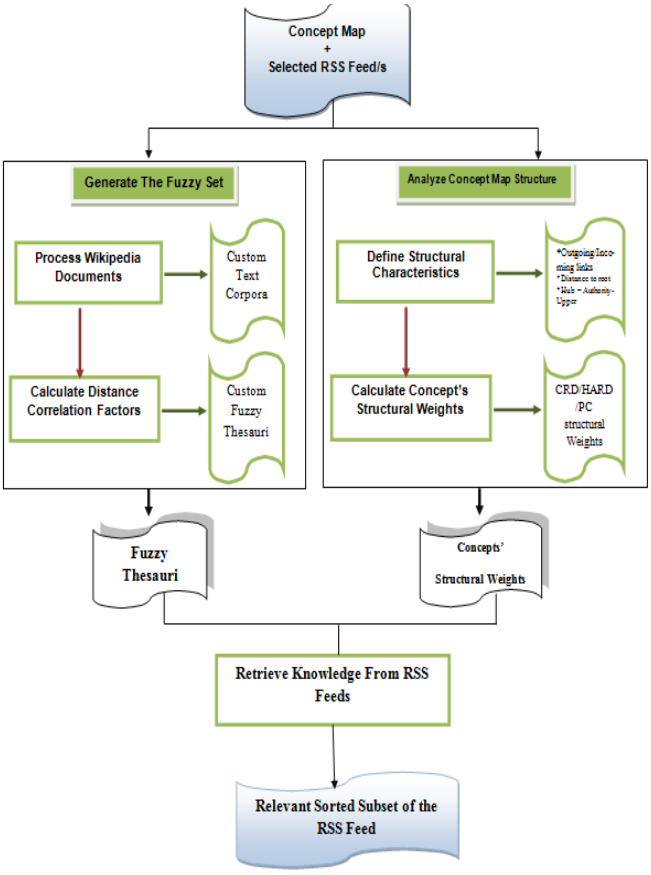

Fig. 3 High Level Definition of the Proposed Method

5.1. Generating The Fuzzy Set – Distance Matrix

As proposed by [14] and later implemented and applied by [15], [18] & [19], Fuzzy Set IR Model starts from fuzzy sets that contain keywords and their relationships which can be thought of as dictionaries or thesauri. Relationship values ranging from 0 to 1 represent the conceptual similarity between two keywords. The conceptual similarity or the fuzzy relationship is also referred as *correlation factors* among words. In this research the distance matrix is used to calculate the correlation factors in the fuzzy set since it's proved in [26] to be the most efficient method over others. Starting from the same concept as proposed by [14], [15], [18] and [19], a set of representative English documents was required to create the distance correlation matrix and hence generate the fuzzy set of keywords. Looking for an unbiased set of documents in terms of writing styles that's diverse in contents and regularly gets updated, *Wikipedia* was selected. *Wikipedia* is a free online encyclopedia, which contains around over 3 millions articles in English as of

2010. *Wikipedia* contains almost all possible topics in different areas of study, is constantly updated, and is unbiased in terms of writing styles and authorship. With the use of the *Wikipedia's* documents to generate the distance correlation matrix, we can obtain reliable similarity measures between different words according to their co-occurrences and distance in various documents. Since in [15] the objective was to generally compare HTML documents, and in [18], [19] the objective was to cluster RSS feeds from different general topics; a comprehensive fuzzy dictionary that contained keywords related to all subjects was required. However, in this research the aim is to provide specific knowledge to specific users where users define their knowledge interests using concept maps. Therefore, in order to enhance the results of the proposed knowledge retrieval method, a *custom distance matrix* is developed for every single concept map. By doing this there will be less noise (i.e. *irrelevant words that specifically and uniquely belong to other fields*) in the distance matrix and hence it's more likely that irrelevant knowledge will be ignored and discarded. The fuzzy dictionary is created through the following two sub-processes:

(i) The process of creating custom text corpora from Wikipedia
(ii) The process of calculating distance correlation factors and creating custom fuzzy thesauri is presented.

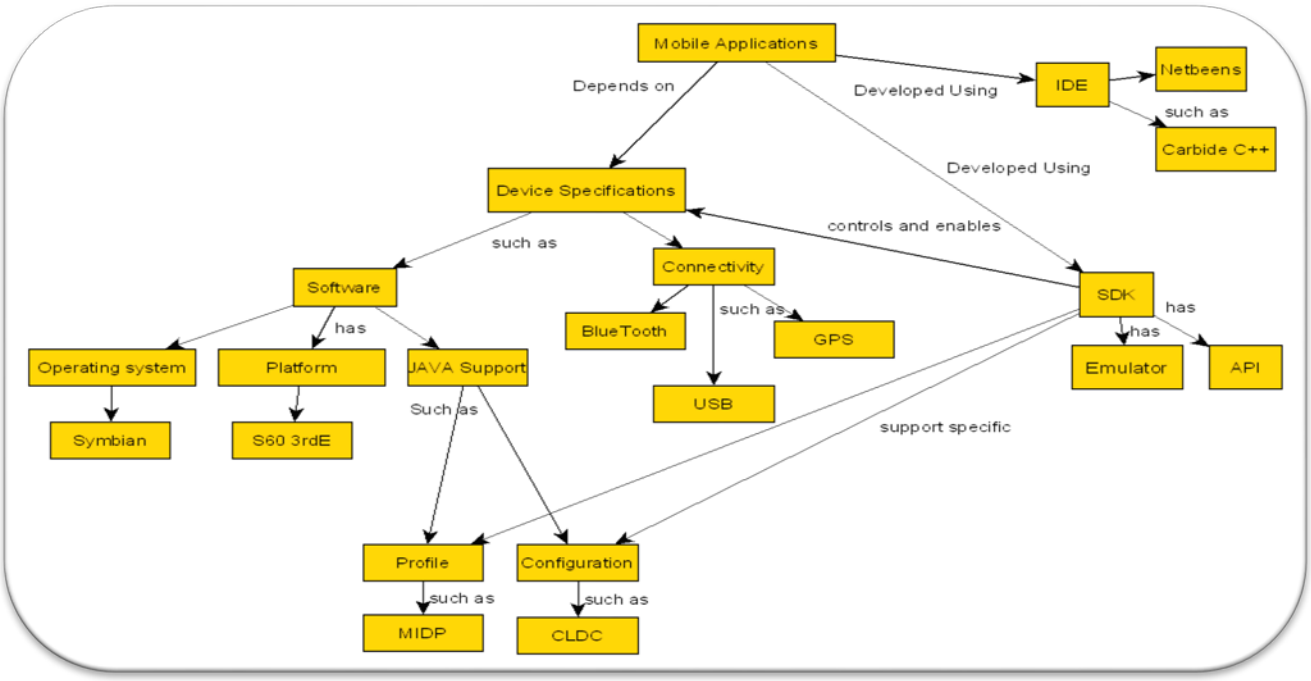

Fig.4 Mobile Applications concept Map

In order to explain the proposed process through a working example a concept map about *"What you need to know when you develop a mobile application?"* was developed with an actual mobile application developer who works on Symbian and uses Netbeens and Carbide C++. The concept map is shown in Fig.4.

5.1.1. Processing Wikipedia Documents – Create Custom Text Corpora

In order to create a custom fuzzy thesaurus, a custom text corpora is required. The process of exporting Wikipedia articles specific to the user's concept map in order to create the custom text corpora, is composed of the following steps as illustrated in Fig.5 :

(i) Read every concept in the concept map
(ii) Perform text checking to deal with multi-term concepts.
(iii) Use Wikipedia Search [1] tool to search the encyclopedia for relevant articles.
(iv) Retrieve the top 1,2,3,4 or 5 articles' titles from Wikipedia search results.
(v) Pass the retrieved titles to Wikipedia Export[1] utility to retrieve corresponding full articles in XML format.

[1] ***ttp://en.wikipedia.org/w/index.php?title=Special:Search***

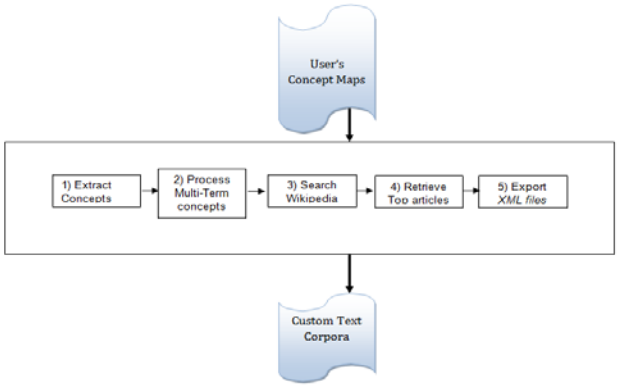

Fig. 5 Creating Custom Text Corpora

5.1.2. Calculating Distance Correlation Factors – Create Custom Fuzzy Thesauri

After a custom text corpora is obtained, it's processed in order to generate a custom fuzzy thesauri. The process of creating custom fuzzy thesauri is shown in Fig.5.4.

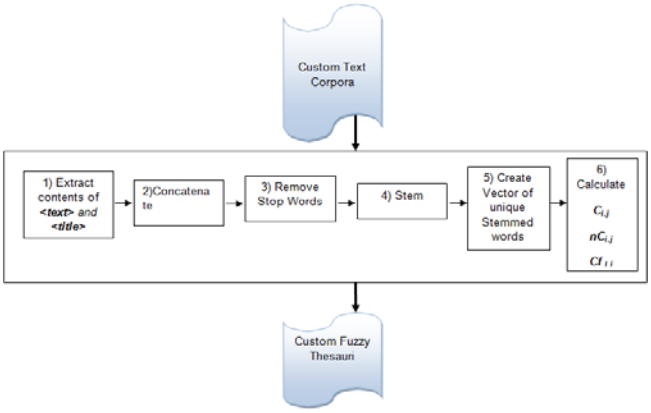

Fig. 6 Creating Custom Fuzzy Thesauri

The *Wikipedia* documents resulted from the export process are then processed in order to extract the contents of the *<title>* and *<text>* tags from the corresponding XML document and concatenate them. Then, all common English StopWords that don't add to the meaning are removed such as propositions and common words (e.g. "The", "about", "above", "across", "after"… etc). Since words that are derived from a single root usually share a common meaning, similarly stemming is used here to group words that share a same stem into semantically related sets. The stem doesn't have to be the same as the root of the word; it is usually enough that related words map to the same stem, even if this stem is not by itself a valid root. Porter stemming algorithm was very commonly used and became the de-facto standard stemming algorithm used for automatic English language stemming. All articles are stemmed using Porter Stemming Algorithm [16]. The Porter stemming algorithm (or 'Porter stemmer') is a process for removing term suffixes such as -ED, -ING, -ION, TION. Furthermore, the suffix removal process will reduce the total number of terms in the custom text corpora, and hence reduce the size and complexity of the data in the method, which is always useful. Stemmed articles are then used to generate vector of all distinct stemmed words along with the documents they appear in and their positions in every document. For example "mobile" stem is "mobil", vector's entry for the word "mobil" would be as illustrated in Table 1.

Table 1 Sample Vector Entry

Term	Document ID	Position
Mobil	Doc1	1, 8, 20, 45
	Doc2	12, 30
	Doc3	7, 10, 27
	Doc4	30

Using the vector of distinct words, (i) the frequency of co-occurrence and relative distance in a single document, i.e., Ci,j, (ii) the normalized value, i.e., $nC\ i,j$, and finally (iii) the distance correlation factor, i.e., $Cf\ i,j,$ can be defined for every pair of keywords across all documents as Eq.(5), Eq.(6) and Eq.(7) respectively:

$$C_{i,j} = \sum_{x \in V(wi)} \sum_{y \in V(wj)} \frac{1}{d(x,y)} \quad (5)$$

$$nC_{i,j} = \frac{C\ i,j}{|V(wi)| \times |V(wj)|} \quad (6)$$

$$Cf_{i,j} = \frac{\sum_{m=1}^{k} nCij}{k} \quad (7)$$

where *d(x, y) = |Position(x) - Position(y)| + 1* is the distance, i.e., the number of words, between words *x* and *y* in a single Wikipedia document, V (wi) and V (wj) is the set of all stemmed variations of words wi & wj in a single Wikipedia document, |V (wi)| & |V (wj)| is the number of words in V (wi) & V (wj) respectively, and *m* is the mth out of the *k* Wikipedia documents in which both wi and wj occur where ($1 \leq m \leq k$). As a result a matrix of all distinct stemmed words and their relationships i.e. *cf,* is constructed. This matrix is the custom fuzzy thesauri which will be used to measure the partial similarity and exact match between concepts in the concept map and terms in the RSS feed.

[1] ***http://en.wikipedia.org/wiki/Special:Export***

5.2. Analyzing Concept Map's Structure

Once the custom fuzzy thesauri is created to handle the semantic similarities, the structure of the concept map is analyzed to measure the relative importance of different terms in the user's concept map. The analysis of the structural weights goes through two steps. *First*, the structural characteristics of each concept need to be defined as per the selected model presented earlier. For CRD model, each concept needs to be characterized for its connectivity and direct steps from the root concept. For HARD model each concept needs to be characterized as being a *hub* with mostly outgoing connections, *authority* with mostly incoming connections or *upper* node that is closer to the root node. For PC model the number of direct and indirect paths from root concept needs to be noted. *Second*, using the structural characteristics the relative concept's weight is calculated.

5.2.1. Defining Structural Characteristics

At this stage the concept map is analyzed and structural characteristics of each node is defined as per every model, i.e. CRD, HARD, PC. For the CRD model the following characteristics need to be defined for every concept:

(i) Outgoing Connectivity (o): number of outgoing connections or links.
(ii) Incoming Connectivity (i): number of incoming connections or links.
(iii) Distance (d): direct steps distance from the root concept.

For example in the concept map in Fig.4, the concept "*SDK*" is one step away from the root, hence, it has a distance of d=1, and connectivity of o=4, and i=1. In the PC model a concept is characterized by the number of paths crossing the concept starting from the root concept. For example concept "*SDK*" has path count of one, whereas concept "***Configuration***" has a path count of three. Finally in the HARD model, concepts are characterized as hub, authority and upper nodes. We are using HITS iterative algorithm was adapted by [11] to calculate the relative hub, authority and upper nodes' positional weights. [11] proved that the proposed algorithm produces positional weights which are ensured to reach a fixed point after a number of iterations equivalent to the number of concepts in the corresponding concept map.

Table 2 Topological Weights after Normalization

Concept	CRD Weight	HARD Weight	PC Weight
Mobile Applications	1.00000	0.92828	0.33333
Device Specifications	0.37037	0.99606	0.66667
Software	0.34568	0.99606	0.66667
Connectivity	0.34568	0.99606	0.66667
operating method	0.09259	0.99605	0.66667
platform	0.09259	0.99605	0.66667
java support	0.17593	0.99606	0.66667
profile	0.13580	0.99605	1.00000
configuration	0.13580	0.99604	1.00000
CLDC	0.00926	0.13860	1.00000
MIDP	0.00926	0.13860	1.00000
USB	0.00926	0.13858	0.66667
bluetooth	0.00926	0.13854	0.66667
GPS	0.00926	0.13846	0.66667
SDK	0.85185	1.00000	0.33333
IDE	0.35185	0.99869	0.33333
Emulator	0.01235	0.13655	0.33333
API	0.01235	0.13534	0.33333
NetBeen	0.01235	0.13230	0.33333
Carbide c++	0.01235	0.12819	0.33333
S60 3rdE	0.00741	0.12259	0.66667
symbian	0.00741	0.11392	0.66667

5.2.2. Calculating Concepts' Structural Weights

After defining the structural characteristics of every concept in the concept map using the three different models, i.e. CRD, HARD and PC. Now the concept weight that reflects its importance in the mind of the user can be calculated as follows:

(i) For CRD Model:

$$W(k) = \left(\alpha \cdot o(k) + \beta \cdot i(k)\right) \cdot \left(1/(d(k)+1)\right)^{1/\delta} \quad (8)$$

(ii) For HARD model:

$$W(k) = \alpha \cdot h(k) + \beta \cdot a(k) + \gamma \cdot u(k) \quad (9)$$

(iii) For PC Model the path count reflect the concept's weight.

After calculating the weights they need to be normalized in order to fall in the interval [0,1]. To do so every concept weight is divided by the maximum weight for the corresponding topological model for the current concept map. For example if in a concept map using CRD model the highest weight was five for a given concept, then all concept weights are divided by five. Weights along with similarity factors are combined later in section 5.3 to retrieve relevant knowledge for users from RSS feeds. See Table 2 for example on the topological weights calculated for the mobile applications concept map.

5.3. Retrieving Knowledge From RSS Feeds

After defining the fuzzy thesauri and the structural weights, the user is asked to define some RSS Feeds from which he/she wishes to receive updates. And the knowledge retrieval process starts automatically as illustrated in Fig.7.

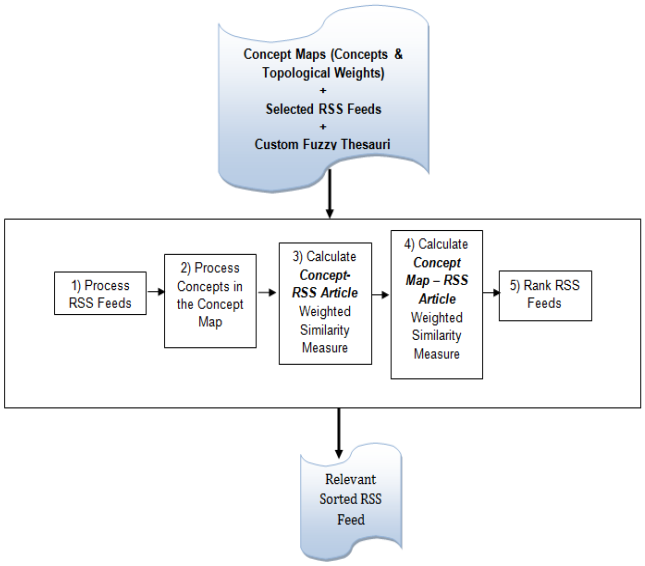

Fig. 7 Process of Retrieving Knowledge From RSS Feeds

These are the steps of the process retrieval process:

(i) Process the RSS feeds as follows:
   a. Extract plain text from the content of the <title> and <content> tags of the RSS feed
   b. Concatenate the content of the <title> and <content>
   c. Remove stop words and stem
   d. Insert all processed RSS articles (i.e. called items in XML syntax) in a vector.

(ii) Process Concepts in the concept map by stemming. Porter Stemming algorithm explained earlier is used her to remove the suffixes from every concept in the concept map.

(iii) Compare every RSS article to the concept map and assign a concept-article similarity weight as follows:

Every stemmed *concept*, i.e. k, in the concept map is compared with every stemmed *word*, i.e. w, in a single RSS *article*, i.e. d, against the pre-computed *distance correlation factors*, i.e. *Cf*, as shown in *Fig.8* , and a similarity measure between the single concept, i.e. *k*, and the RSS article, i.e. *d*, is calculated according to Eq.(10).

$$Concept\_Article\ Similarity\ (\mu_{k,d}) = 1 - \prod_{w \in d}(1 - Cf_{kw}) \quad (10)$$

For multiple-term concepts, the distance correlation factors, i.e. *Cf*, for every term is compared separately and then the average distance correlation factors is computed to be used in Eq.(10). For example for the concept "Mobile applications", the proposed method would look up for $Cf_{mobil,w}$ and $Cf_{applic,w}$ and use the average value of both correlation factors.

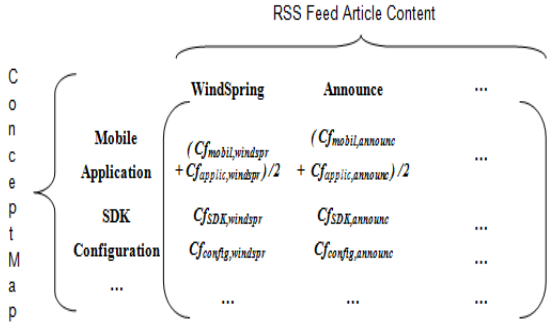

Fig. 8 Concept-RSS Article Comparison

(iv) The similarity measure is then multiplied by the concept's structural weight i.e. *W*, calculated earlier in section 5.2.2. A weighted similarity measure is produced using Eq.(11):

$$Weighted\ Concept\_Article\ similarity = W(k) * \mu_{k,d} \quad (11)$$

(v) The overall weighted similarity between the concept map and a single RSS feed article i.e., *d*, is calculated according to Eq.(12).

$$Weighted\ ConceptMap\_Article\ similarity = \frac{W(k1).\mu k1,d + W(k2).\mu k2,d + \ldots + W(kn).\mu kn,d}{n} \quad (12)$$

Where *n* is the number of concepts *k* in the concept map.

(vi) Repeat steps (i) through (v) until all RSS articles are compared to the concept map

(vii) Display the top 5, 10, 15 or 20 RSS articles to the user in a new RSS feed.

6. Implementation

For the sack of validation of the proposed method, a JAVA program was developed. Eclipse was used as a development environment with JAVA Runtime Environment (JRE) version 6. yFiles library [27] was used for graph, i.e. concept map, parsing and analysis. Lucene library [28] was used for applying the porter stemming and stopWords removal algorithms. Wikixmlj library [29] was

used to parse Wikipedia files in xml format. Rome library [30] was used for parsing RSS feeds. The program was tested on a Linux machine so as to allow for the maximum size of the JAVA Virtual Machine Memory (JVM).

7. Evaluation

In order to evaluate the proposed method multiple tests were conducted.

(i) *First*, the fixed parameters in models HARD and CRD were tested to find out how sensitive the method results were to those parameters and whether different values of the parameters would affect the usefulness of the results generated by the method. A *factorial sensitivity analysis* was conducted to assess the parameters. In the factorial analysis the Mean Square Error (MSE) between Google feed's rank and the proposed method's rank was used as a response variable. Four factorial experiments were conducted to test different values of the fixed parameters α, β and δ at the 0.05 level.

(ii) *Second, relevance and precision* of the results produced by the method were tested with the actual user of the test concept map using Eq. (1) and Eq. (2) mentioned earlier. The user was asked to select top five related articles in multiple tests, and then user's selection was compared to method's selection to see to what degree the method matches the user's knowledge interests. Throughout the evaluation process, the test concept map about mobile application development presented in Fig.4 was used. User of the concept map is working in the field of mobile application developments.

8. Results and Discussion

In the factorial sensitivity analysis, the p-value indicates that there is no significant effect of any of the parameters used in the models HARD and CRD. As a result, changing the values of the parameters α, β and δ doesn't significantly change the rank generated by the method. This in turn, indicates that the method performance is stable using different parameters values and consequently the major influence on the method's result comes from the similarity factors and the topological weights of the concepts in the concept map rather than method's parameters. Therefore, the method's results are more likely to specifically reflect the concept map's context without being influenced by the method's parameters.

The proposed method exhibited high ability in selecting relevant articles from single and multiple RSS feeds. It also successfully excluded noise or irrelevant knowledge that was not interesting to the user. When applying the method to RSS feeds with number of articles between 10 and 20, the method successfully and repeatedly over several tests selected top three articles exactly as selected by the user of the concept map. The remaining 2 articles selected by the method were found interesting most of the time by the user. In 2 of the tests precision value of the generated results was 100%, where it was 80% in the other tests. The method repeatedly never selected any of the articles that were highlighted by the user as very uninteresting.

9. Conclusion, Limitations And Future Work

A typical web surfer likes in every web search to have every result on the first page to be relevant. Considering RSS feed as a source of newly created knowledge on the web the proposed method attempts to combine the semantic and structure of concept maps along with the logic of Fuzzy Set IR Model to update users of the web on relevant knowledge as it becomes available. Experimental results show that the proposed method is consistent in retrieving relevant knowledge for a single concept map. The partial factorial experiment conducted on two replicas shows that the method's results are not significantly affected by different parameters values, hence, results are more influenced by the semantic and structure of the concept map. Finally in order to test the precision of the results, a user test was conducted at 4 stages with an RSS feed relevant to mobile application development, an RSS feed generally related to technology, an RSS feed generally related to software development and an irrelevant RSS feed. Precision in the four stages ranged between 100% and 80% which is relatively very high. In this research the objective of instantly updating web users on newly created knowledge that is tailored to their specific interests was accomplished as shown by the experimental results. However, the proposed method by no means exhausted the subject. Further efforts can enhance the precision and the performance of the proposed method:

(i) *User Interest Definition*: using concept map as a tool to define knowledge interest proved to be effective as shown by the research results. However, if user didn't create good concept map then the proposed method won't be as precise in retrieving relevant knowledge. Some efforts can be made to help users create good concept maps.

(ii) Sources of Knowledge: the proposed method can be generalized to account for other sources of knowledge such as forums, wikis and blogs.

References

[1] David Grossman, Ophir Frieder. "Retrieval Startegies" in *Information Retrieval Algorithms and Heuristics*, 2nd ed. Netherlands: Springer, 2004, pp. 9-91.

[2] Watraru Sunayama, Yukio Osawa, Masahiko Yachida. "Serach Interface for Query Restructuring with Discovering User Interest" , in *IEEE Third International conference on Knowledge-Based Intelligent Information Engineering Systems*, Adelaida, Australia: IEEE, 1999 , pp.538-541.

[3] David Leak, Ana Maguitman, Thomas Reichherzer. "GOOGLING" From a Concept Map: Towards Automatic Concept-Map-Based Query Formation", in *Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain: Universidad Pública de Navarra, 2004, pp. 409-416.

[4] Imran A. Zualkernan, Mohammed A. AbuJayyab, Yaser A. Ghanam, "An Alignment Equation for Using Mind Maps to Filter Learning Queries from Google" , *in Sixth IEEE International Conference on Advanced Learning Technologies (ICALT'06)*, The Netherlands, 2006, pp. 153-155.

[5] Ana Gabriela Maguitman, David B. Leake, Thomas Reichherzer, Filippo Menczer. "Dynamic Extraction Topic Descriptors and Discriminators: towards automatic Context-Based Topic Search" *ACM-CIKM International Conference on Information and Knowledge Management*, Washington DC, USA, 2004, pp. 463-472

[6] Brian, Kelly. "RSS - More Than Just News Feeds" *New Review of Information Networking*, vol. 11, 2005

[7] CNN.com, *http://rss.cnn.com/rss/cnn_latest.rss*, accessed on 21/Oct.2010

[8] RSSOwl, *http://www.rssowl.org*, accessed on 21/Oct.2010

[9] Google SOAP Search API , *http://code.google.com/intl/ar/apis/soapsearch/reference.html*, accessed on 1/10/2010

[10] Joseph Novak, Alberto Canas, "The Theory Underlying Concept Maps and How to Construct Them", *Technical Report IHMC CmapTools 2006-01*, Florida, USA: Florida Institute for Human and Machine Cognition (IHMC), 2006.

[11] Alberto J. Cañas, David B. Leake, Ana Gabriela Maguitman. "Combining Concept Mapping with CBR: Towards Experience-Based Support for Knowledge Modeling", *in Fourteenth International Florida Artificial Intelligence Research Society Conference.* Florida , USA: AAAI Press, 2001, pp.286 - 290.

[12] David Leak, Ana Maguitman Thomas Reichherzer. "Understanding Knowledge Models : Modelling Assesment of Concept Importance in Concept Maps", *in 26th Annual Conference of the cognitive Science Society*, New Jersey: Lawrence Erlbaum, 2004, pp.785-790.

[13] Amy Langville, Carl Meyer. "the HITS Method For Ranking Web Pages" in *Googlee's Page Rank and Beyond The Science of Search Engine Rankings.* 1st ed. Princeton University Press, 2006, pp. 115-126.

[14] Yasushi Ogawa, Tetsuya Morito, Kiyojiko Kobayashi. "A Fuzzy Document Retrieval System Using The Keyword Connection Matrix and a Learning Method", *in Fuzzy Sets and Systems*, 39(2), 1991, pp. 163-179.

[15] Rajiv Yerra, Yiu-Kai Ng. "Detecting Similar HTML Documents Using a Fuzzy Set Information Retrieval Approach", *in Granular Computing, 2005 IEEE International Conferenc*, 2005, pp. 693 - 699.

[16] Porter., Martin. "The Porter Stemming Algorithm", Jan 2006, in *Martin Porter's Home Page*, accessed in April 2010, url: <http://tartarus.org/~martin/PorterStemmer/>.

[17] Ng, Chris Tseng and Patric. "Precisiated Information Retrieval for RSS Feeds", in *Information Management & Computer Security*, 15(3), 2007, pp. 184-200.

[18] Nathenal Gustafson, Maria Pera and Yiu-Kai Ng. "Generating Fuzzy Equavelace Classes on RSS News Articles for Retrieving Correlated Information" , in *International Conference on Computational Science and Applications*, Berlin, Heidelberg: Springer-Verlag, 2008, pp.232-247.

[19] Maria Pera and Yiu-Kai Ng. " Utilizing Phrase-Similarity Measures For Detecting And Clustering Informative RSS News Articles" , *in Integrated Computer-Aided Engineering*, vol. 15, 2008, pp. 331–350

[20] C. Demaio, G. Fenza, V. Loia and S. Senatore. "Ontology-Based Knolwedge Structuring: An Application on RSS Feeds", i*n IEEE - 2nd International Conference on Human System Interaction*, Catania, Italy, 2009, pp.467-473.

[21] Christopher Manning, Prabhakar Raghavan. "Ch1, Ch6, Ch8, Ch21" *in Introduction to Information Retrieval*, USA: Campridge University Press, 2008.

[22] Amit, Singhal. "Modern Information Retrieval: A Brief Overview", *IEEE Data Engineering Bulletin*, vol. 24, pp. 35-43, 2001

[23] C. J. van Rijsbergen, "Chapter 1" , in *Information Retrieval* , 2nd ed , Department of Computing Science University of Glasgow, http://akira.ruc.dk/~jv/KIIS2004/ir1_2.pdf, accessed on April 2010

[24] Thomas Jech, "Set Theory", *published Thu Jul 11, 2002*, http://plato.stanford.edu/entries/set-theory/, accessed on April, 2010.

[25] Zadeh, "Fuzzy Sets.", *in Information and Control* , vol. 8, 1965, pp. 338-353.

[26] Ian Garcia, "Eliminating Redundant And Less-Informative RSS News Articles Based On Word Similarity And A Fuzzy Equivalence Relation", M.Sc. Dissertation, Brigham Young University, 2007

[27] yFiles for Java by *yWorks GmbH*, http://www.yworks.com/yfiles

[28] Apache Lucene by the *Apache Software Foundation* , http://www. lucene.apache.org/

[29] Wikixmlj by *Delip*, http://code.google.com/p/wikixmlj/

[30] ROME by *JAVA.Net*, http://wiki.java.net/bin/view/Javawsxml/Rome

Heba Ismail received her BSc. In Information Systems Technology from Abu Dhabi University in 2007 and worked as software engineer for Emirates Group IT for two years after that she joined the American University of Sharjah as a Graduate Teaching and Research Assistant. She Received her MSc. in Engineering Systems Management with Concentration in IT Management in 2011. She did research on managing knowledge on the web, using fuzzy set theory to enhance quality and precision of retrieved knowledge, using RSS, Blogs and Twitters as components of knowledge management systems and utilizing mind mapping and concept maps to elicit users' knowledge requirements. After obtaining her master degree, she continued working as a research assistant and now she is running her own business providing educational web and mobile solutions for schools.