Available online at www.isas.org.in/jisas

JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 67(1) 2013 97-106

An Intelligent Semantic Search Engine with Cross Linguility Support

S.D. Samantaray

Department of Computer Engineering College of Technology, GBPUA&T, Pantnagar

Received 01 July 2012; Revised 26 November 2012; Accepted 02 December 2012

SUMMARY

In this paper, an intelligent semantic based search engine has been presented that can be used as a multilingual platform for different search queries. It retrieves those results pages also which don't have directly the keywords but contains the synonyms or related words. In response to a query for the word *"soil"* it will also retrieve web pages which don't have directly the word *"soil"* but have the semantically related words such as *"land"*, *"ground"*, *"earth"*, "*loam"* etc. As a cross lingual support it retrieves the pages of other languages which have semantic presence of the searched keyword. These pages are presented to user in the original query language after necessary translation/transliteration. Spell-check support is also provided for giving suggestions in respect of misspelled queries. For the search engine implementation, JSP (Java Server Pages) has been used to integrate the Front-end (webpage) and Back-end (crawler and indexer). LUCENE (a very popular open source information retrieval library) and Word Net library has been used for getting the synonyms list of the queried words.

Keywords: Intelligent search, Natural language processing,Semantic web, Artificial intelligence, Intelligent systems, Cross linguality.

1. INTRODUCTION

The latest development in the field of machine learning and information retrieval boosted the mathematical basis of search engine technologies. Efficient parsing, stop word filtering, stemming, feature extraction, formulizing mathematical model, spelling correction, unsupervised clustering, supervised classification, page ranking based on markov chain and implicit-explicit feedback models are important to mention as limbs of every successful next generation search engines. The personalization of results based on location, time of day and based on recommendation for like-minded users make it very practical.

Information on the internet is present in many languages and no person can understand all the languages. So, this language barrier in accessing the information should be crossed. Keyword search engines do not perform any meaning based search, which means if the user queried for a particular keyword and that keyword is not available directly on the web, the search engine would not return any result. We target to design such a search engine, which would return those pages also which contain the words related to the queried words.

We have developed a keyword based search engine but with certain improvements over the current scenario. Keyword searches have a tough time distinguishing between words that are spelled the same way, but mean something different (*e.g.* a hard stone, a hard exam, and the hard drive on your computer). This often results in hits that are completely irrelevant to your query. Some search engines also have trouble with so-called stemming – *i.e.*, if the entered word is 'big'

E-mail address : sdsray@gmail.com

should it return a hit on the word 'bigger?' What about singular and plural words? What about verb tenses that differ only 's' or 'ed'? Search engines also cannot return hits on keywords that mean the same, but are not actually entered in query. A query on heart disease would not return a document that used the word 'cardiac' instead of 'heart'.

Plain keyword search do not perform meaning or concept based search, which means that if the queried-keyword is not present directly on the web page, the search engine would not return relevant result. For example, a query on "heart" would not return a document that contains the word "cardiac". However, semantic search would return those pages also which contain the words related to the query. In this way, the search would be improved relative to the plain keyword search. Further, search engines don't cross the language barriers for accessing the information over the web. Today, web is the most exciting and the biggest source of the information which is present in all the languages. But, no one can understand all the languages. Crossing the language barrier is still a great challenge. In this paper, an approach for the development of a cross lingual semantic web search engine has been presented. This search engine returns those web pages also which do not contain directly the searched words but they have the synonyms or related words of the queried words. As a cross lingual support it also retrieves the pages of other languages than the users' native language, which have semantic presence of the searched keyword. Moreover, it gives back the results in the user's language even if the resultant web page is in different language. For this it uses machine translation provided by Google translator and Google API for translation/ transliteration purpose.

A semantic search engine attempts to make sense of search results based on context. It automatically identifies the concepts structuring the texts. An important part of this process is disambiguation, both of the queries and of the content on the web through natural language processing, meaning that it can disambiguate between 'car' and 'big cat' for a search of "jaguar". Semantic search seeks to improve search accuracy by understanding searcher intent and the contextual meaning of terms as they appear in the searchable dataspace to generate more relevant results. (Lisa *et al.*, Ramanathan)

2. CROSS LINGUAL AND SEMANTIC

Cross Lingual Information Retrieval (CLIR) is the technology that retrieves information from various documents and database written in language different from those in which key words are written. *e.g.* It could be used to retrieve information from English academic papers or German new articles using Hindi keywords. This task has also been termed multilingual, translingual, or cross-language Information Retrieval. CLIR is especially important to countries where a very large fraction of the population is not conversant with English and consequently does not have access to the vast store of information that is available in English on the Internet. Using a CLIR system, a single query could retrieve documents irrespective of the language. The collection contains multilingual documents with text in two or more languages. (Lisa *et al.*, Ramanathan)

3. ACHIEVING CROSS LINGUALITY

Cross Lingual Information Retrieval is a challenging task typically involving query translation into multiple languages followed by monolingual retrieval in each language. Following are the main approaches to achieve it:

3.1 Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA) is a theory and method for extracting and representing the contextual-usage meaning of words by statistical computations applied to a large corpus of text. The underlying idea is that the aggregate of all the word contexts in which a given word does and does not appear provides a set of mutual constraints that largely determines the similarity of meaning of words and sets of words to each other. The adequacy of LSA's reflection of human knowledge has been established in a variety of ways. For example, its scores overlap those of humans on standard vocabulary and subject matter tests; it mimics human word sorting and category judgments; it simulates word – word and passage – word lexical priming data; and, it accurately estimates passage coherence, learnability of passages by individual students, and the quality and quantity of knowledge contained in an essay. Latent Semantics Analysis (LSA) allows cross-lingual retrieval without translating queries by working from an already existing corpus. It obviates

the need to construct complicated translation tools, making this technique particularly applicable to querying less commercially appealing language. LSA involves constructing a term document matrix for a large collection of documents. This matrix gives the number of occurrences of each term (which are essentially the same as words) within each document. Singular value decomposition is then used to construct the semantic space for the corpus. LSA assumes that there is some amount of noise present in all natural language due to flexibility in expressing the same concept. It attempts to remove this noise and represents the underlying concepts within documents via re-multiplication of the decomposition matrices using a reduced number of factors (Landauer Thomas *et al.* 1998, Lawrence *et al*.). Perhaps the greatest benefit of LSA is its ability to overcome the fundamentals problems of *Synonymy*, *Polysemy* and *Inflexion* which are inherent in natural language processing. *Synonymy* means that many different words have similar meaning. For example, if we searched for the word *"large"* then the keyword matching techniques would not retrieve relevant documents containing the words *"big", "huge"* or *"massive"* etc. However, LSA recognizes that these words all refer to the same concept and hence would retrieve all of these documents. A *Polysemous* word is one, which has several meanings depending on the context. For example, the word *"chip"* could be referring to *fish and chips*, *a computer chip*, *a gambling chip* or *a chip of wood*, depending on the context. *Inflexion* is the process of addling affixes to or changing the base form of a word. The words '*doing*', '*did*' , '*does*' , and '*done*' are all related to "*do*" but would not be retrieved by keyword-matching techniques upon searching for this word without the use of sonic form of stemming. While stemming techniques vary in accuracy, the majorities are based on rules of grammar and are therefore not cross lingual. Only co-occurrence based stemmers can be applied over multiple languages.

Although LSA has many advantages, there are also some limitations. Firstly, LSA is hampered by the processing power and memory capacity of computers. Empirical results suggest that the larger the size of the training set, the better the performance of LSA. However, constructing a term-document matrix for a relatively small set of 2000 documents can take up to an hour and use 300Mb of memory. It is hoped that due to the continual increase in the speed and memory of computers this will not be such a problem in the future.

Secondly, LSA is a "bag of words" technique which means that it makes no use of word order. Hence, we expect it to miss some of the concepts within documents. Thirdly, LSA is limited by the size of the text corpora used. Apart from taking a long time to process, sufficiently large text corpora simply may not be readily available for specific domains.

3.2 Explicit Semantic Analysis (ESA)

Explicit Semantic Analysis (ESA) attempts to index or classify a given text t with respect to a set of explicitly given external categories. It is in this sense that ESA is explicit compared to approaches which aim at representing texts with respect to latent topics or concepts, as done in Latent Semantic Analysis (LSA). A very interesting characteristic of Wikipedia, besides the overwhelming amount of information created dynamically and in a collaborative way, is the fact that articles are linked across languages. Cross-language links are those links that an article has corresponding to the article in the other language. A previous analysis of this cross-lingual link structure between the German and English Wikipedia showed that 95% of these links are indeed bi-directional. The analysis of French-English and French-German links showed similar results. We therefore assume the existence of a mapping function $m(i \to j)$ that maps an article of W_i in one language to its corresponding article W_j of other language. (Philipp and Philipp). In fact, given a text $t \varepsilon T$ in language L_i, it turns out that we can simply index this document with respect to any of the other languages L_1,.., L_n we consider by transforming the vector $\Phi_i(t)$ into a corresponding vector in the vector space that is spanned by the articles in the target language. Thus, given that we consider n languages, we have n^2 mapping functions. In order to get the ESA representation of a document $t \varepsilon T$ in language L_i with respect to article W_j of other language we simply have to compute the necessary function. Thus it gives an elegant retrieval model which is uniform across languages. A prerequisite for this model is certainly that we know the language of the query and of the different documents in order to know which mapping should be applied.

3.3 Universal Networking Language (UNL)

The Universal Networking Language (UNL) been introduced as a digital meta language for describing, summarizing, refining, storing and disseminating information in a machineindependent and human-language-neutral form. The UNL represents information, *i.e.*, meaning, sentence by sentence. Sentence information is represented as a hyper-graph having concepts as nodes and relations as arcs. This hyper-graph is also represented as a set of directed binary relations, each between two of the concepts present in the sentence. Concepts are represented as character-strings called *Universal Words (UWs)* (Mukerjee *et al.* 2003, Nguyen Dat and Ishizuka, Bhattacharya). Universal Networking Language (UNL) is an artificial computer language representing information or knowledge conveyed by natural language texts. It comprises of UNL- Expressions that have Universal Words, Constraint List, Relations and Attribute labels. UNL-Expressions are unambiguous with support for semantic analysis.

Universal Networking Language (UNL) vocabulary consists of:

- **Universal Words (UWs):** Universal Words are words that constitute the vocabulary of UNL. A UW is not only a unit of the UNL syntactically and semantically for expressing a concept, but also a basic element for constructing a UNL expression of a sentence or a compound concept. Such a UW is represented as a node in a hypergraph. There are two classes of UWs from the viewpoint in the composition:
  - labels defined to express unit concepts and called "UWs" (Universal Words).
  - a compound structure of a set of binary relations grouped together and called "Compound UWs".

A UW is a English-language word followed by a list of constraints. (Nabab *et al.* 2011) The following is the syntax of description of UWs in Context Free Grammar (CFG):

```
<UW> :: = <headword> [<constraint list>]
<headword> :: = <character> ...
<constraint list> :: = "("<constraint> [","  <constraint>] ... ")"
<constraint> :: <relation label> {">" | "<"} <UW>
<constraint list>] ...
<relation label> {">" | "<"} <UW> [<constraint list>]
[{">" | "<"} <UW> [<constraint list>]]...
<relation label>:: = "agt" | and" | "aoj" | "obj" | "icl" | ...
```

- **Relation Labels**: These are the tags that represent the relationship between Universal Words. The relation between UWs is binary that have different labels according to the different roles they play. A relation label is represented as strings of three characters or less. The following is an example of relation defined according to the above principles.

  Relation: agt (agent)
  agt defines a thing that initiates an action. agt (do, thing) agt (action, thing) Syntax: agt [" : " < Compound UW-ID >] " (" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

  An agent is defined as the relation between: UW1 - do, and UW2 - a thing .

  Here UW2 initiates UW1, or UW2 is thought of as having a direct role in making UW1 happen.

- **Attribute Labels:** They express additional information about the Universal Words that appear in a sentence. The attributes represent the grammatical properties of the words. Attributes of UWs are used to describe subjectivity of sentences. They show what is said from the speaker's point of view: how the speaker views what is said. This includes phenomena technically called speech, acts, propositional attitudes, truth values, etc. Conceptual relations and UWs are used to describe objectivity of sentences. Attributed of UWs enrich this description with more information about how the speaker views these states-of-affairs and his attitudes toward them. For example, the corresponding UW of play is "play (icl>do)". If the word "play" is in the past form in the sentence an attribute @past is tagged with "play (icl>do)". If it is the main word in the sentence then @entry will be tagged such as "play (icl>do),@entry, @past".

- **UNL Expression:** An UNL Expression can be seen as a UNL graph. The UNL expresses information or knowledge in the form of semantic network. UNL semantic network is made up of a set of binary relations, each binary relation is composed of a relation and two UWs that hold the relation. A binary relation of UNL is expressed in the following format:

  <relation> (<uw1>, <uw2>)
  In <relation>, one of the relations defined in the UNL specifications is described. In <uw1> and <uw2>; the two UWs that hold the relation given at <relation> are described.

- **Hypergraph:** The UNL expression is a hyper semantic network. That is, each node of the graph, <uw1> and <uw2> of a binary relation, can be replaced with a semantic network. Such a node consists of a semantic network of a UNL expression and is called a "scope". A scope can be connected with other UWs or scopes. The UNL expressions of in a scope is distinguished from others by assigning an ID to the <relations> of the set of binary relations that belong to the scope. The general description format of binary relations for a hyper-node of UNL is the following:
  <relation>:<scope-id> (<node1>, <node2>)
  where, <scope-id> is the ID for distinguishing a scope. <node1> and <node2> can be a UW or a <scope node>. A <scope node> is given in the format of ":<scope-id>".

- **Knowledge Base:** The UNL Knowledge Base gives possible binary relations between UWs. The knowledge base is a set of knowledge base entries. The format of knowledge base entries is as follows.

<Knowledge Base entry> :: = <Binary relations> "=" <degree of certainly><Binary Relation> :: = <Relation Label> "("<UWl>", "<UW2>")"<degree of certainty> ::= "0" | "1" |...| "255"

When the degree of certainty is "0", it means the relation between two UWs is false. When the degree of certainty is more than "1", it means the relation between two UWs is true, and the bigger the number is, the more the credibility is UNL was developed as a universal knowledge-encoding mechanism, and is being primarily driven by the needs of the Machine Translation community. UNL provides for a uniform concept vocabulary (called "universal words" or UW's – the same concept in any language results in the same UW, which is written out using English orthography). These UW's are connected by a small set of about thirty-eight binary relations to obtain a set of predicate expressions that can encode the linguistic content of any sentence in any language of the world (Bhattacharya, Alansary *et al*.). Fig. 1 depicts the core architecture of the UNL system.

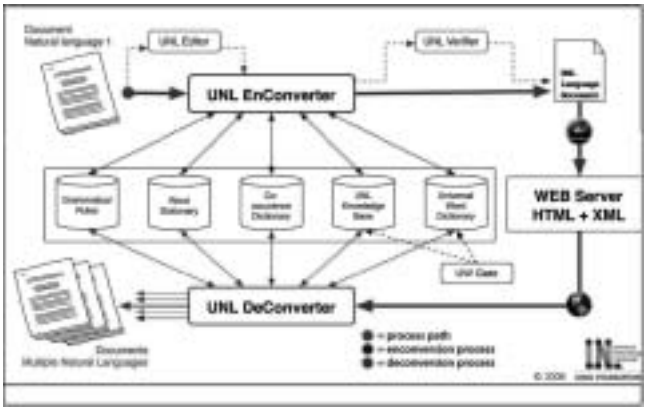

Fig. 1: The core architecture of the UNL system

The UNL may be thought to be an inter lingua, but UNL has a number of other features that make it better suited for semantic inference than most other interlinguas. In particular, the following features of UNL motivate this work:

- The set of Universal Words with well defined universal interpretations.
- A small, simple predicate structure with only binary predicates.
- A knowledge base connecting the UW's as a weighted graph of relations.
- Ontological information that is built-in to the UWs (eg. Khaira (icl<disease) characterizes Khaira as a type of disease).
- The world wide effort in developing mechanisms for converting language into UNL and vice versa.
- The dream of language independent semantic analysis.

Enconverter is used for converting Source natural language sentences to UNL-expressions and

Deconverter are used to generate Target Language sentences from UNL- expressions. The schematics of Enconverter and Deconverter are given in Fig. 2 and Fig. 3 respectively. An Enconversion Rule (morphological/semantic) is composed of conditions for the nodes placed on the Analysis Windows and Condition Windows, and Actions and/or Operations for the nodes placed on the Analysis Windows. Such enconversion rules describe the kind of actions and/or operations that should be carried out for all phenomena of a language, and under what conditions. EnConverter will find the most suitable rule every time, and create a partial syntactic tree and/or UNL expression. A set of UNL expressions of a sentence will finally be completed after having applied a set of all the necessary rules.

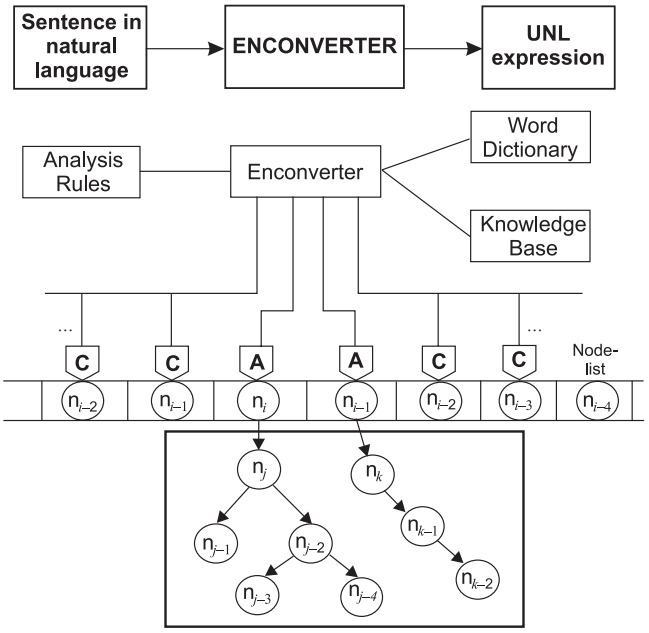

Fig. 2: Structure of EnConverter "A" indicates an Analysis Window, "C" indicates a Condition Window, and "n_n" indicates an Analysis Node

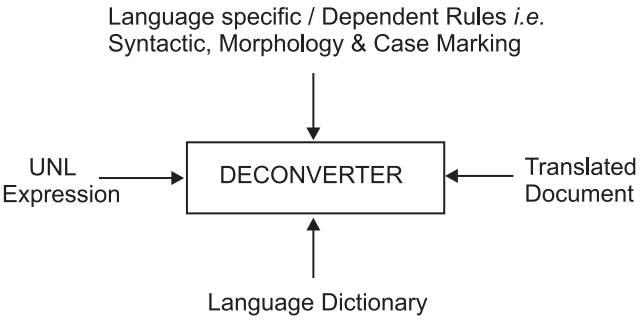

Fig. 3: Structure of Deconverter

4. DESIGN AND IMPLEMENTATION

Fig. 4 shows the basic working schema of the search engine while Fig. 5 presents detailed implementation diagram. The user puts the query in own native language. The language of the keyword / query is converted to the language of the index available. The relevant documents are retrieved through these indices and the retrieved documents are translated to the user's language.

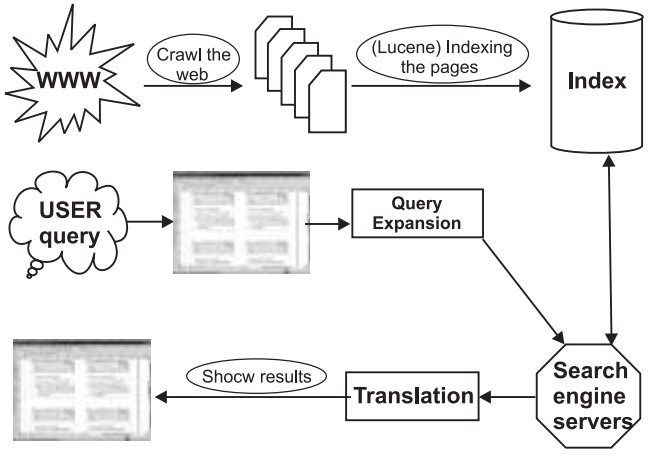

Fig. 4: Basic working schema

It is a java based implementation with the use of JSP (Java Server Pages) for integrating the Front-end (webpage) and Back-end (crawler and indexer). The searcher code and the interface design are both

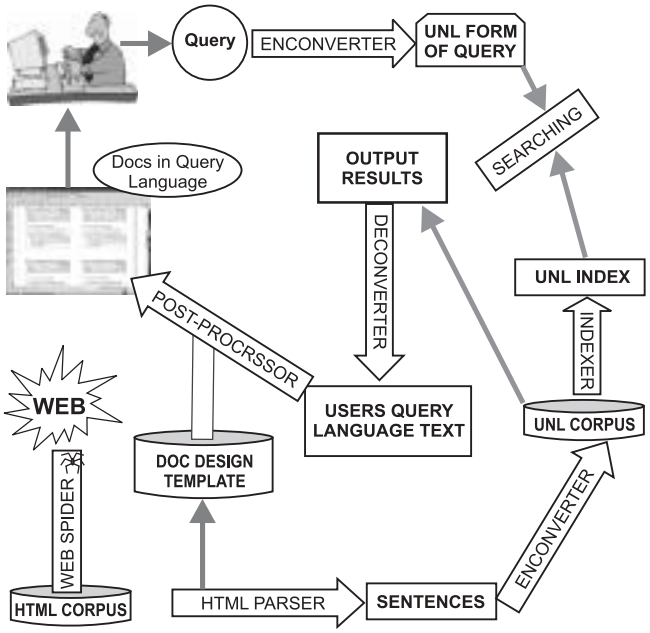

Fig. 5: Detailed Implementation Diagram

embedded in the Java Server Pages. LUCENE (a very popular open source information retrieval library) has been used to implement indexer and searcher. WordNet library has been used for getting the synonyms list of the queried words.

The various modules used in the implementation has been shown in Fig. 6. It shows both the perspectives simultaneously. On the one hand, the back end is running i.e. the crawler runs over the web to collect the HTML pages which are indexed. And on the front end, the user puts a query which goes to the search engine servers after the query expansion (it adds the semantically related words to the query). Search Engine servers take the query and search through the index. Finally, these results are translated to the user's language before giving them back to the user.

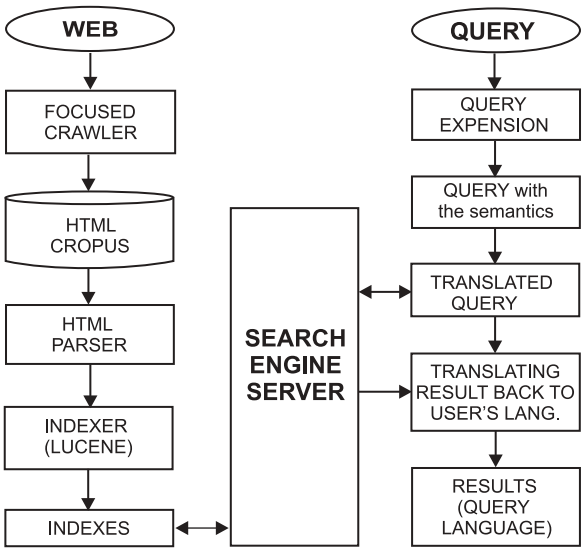

Fig. 6: Various modules of Search Engine

This section describes the overall structure, data and control flow of the search engine.

4.1 Crawler

We implemented a multi-threaded crawler so that it can download several pages simultaneously and the indexer be fed the pages nonstop. Following are the steps to run a crawler that fetches all possible pages from an initial set of pages:

1. Load the initial set of URLs into a list of pages to be indexed.
2. Run till completion
   (a) Check the validity of the URL, then check the *robots.txt* for ethical crawling, then find an unindexed page, fetch and store the page, extract the outgoing links and add to the list of pages to be indexed
   (b) Exit the loop, if a time limit for the crawler is exceeded or there are no more un-indexed pages left.

For each page p and each visit, the following information is available:

- The access time-stamp of the page.
- The last-modified time-stamp (given by most Web servers; about 80%-90% of the requests in practice)
- The text of the page, which can be compared to an older copy to detect changes.

Different crawlers may follow different crawling strategies to make crawling faster and more efficient. For example, the Google spider was built to index every significant word on a page, leaving out the articles 'a', 'an' and 'the' called "stop words". Sometimes site owners typically do not welcome visits from crawlers for several reasons. Informal protocols to deny or allow a crawler to visit a site do exist; however, it is left to the crawler's author to observe a site's crawler directives.

4.2 HTML Parser

This module parses the html documents crawled by the crawler in order to separate the formatting (HTML tags) from the sentences. The design of the document is stored in the document design template, which consists of only HTML tags with the placeholders for sentences.

We designed a basic HTML parser for extracting the page contents and the title of the HTML page. And, we made these the values of the fields namely "contents" and "title" of the Lucene documents which are indexed.

4.3 Indexer

Search engine indexing collects, parses, and stores data to facilitate fast and accurate information retrieval. Larger servicers typically perform indexing at a predetermined time interval due to the required time and processing costs, while agent-based search engines index in real time.

This module takes as input the HTML corpus and generates an index of the HTML pages, which is used by the search module to quickly find the relevant documents and calculate their relevance.

We are using Lucene for indexing. Lucene is a powerful search framework capable of indexing a several gigabytes of document data and then quickly performing complex searches on that data. Lucene can also process data beyond raw text. Typically this consists of data about the documents that are being indexed, for example, title information or document authors. Lucene provides a scoring algorithm that includes this additional data to find best matches to document authors. Lucene provides a scoring algorithm that includes this additional data to find best matches to document queries. The default scoring algorithm is fairly complex and considers such factors as the frequency of a particular query term with individual documents and the frequency of the term in the total population of documents. This module preprocesses the HTML documents and converts them to an intermediate form which is then indexed by the indexer. Following steps are performed by this module:

1. The extracted text is put into a newly formed Lucene document into various fields such as “title”, “contents”, “path”, ”summary” etc.
2. Then this lucene document is then indexed or added to the index by the Lucene. Major factors in designing a search engine’s architecture include: Merge factors, Storage techniques, Index size, Lookup speed, Maintenance, Fault tolerance.

4.4 Translation

Statistical machine translation has been used in our project for achieving cross linguality. It produces better results than the other methods. Moreover, it is quite simple to use because Google provides a free API (application programming interface) for machine translation. We are using Google translation web service for getting our translation. This API uses statistical machine translation. Its results are really excellent and have been well tested for a long time on the Google. The API can be easily embedded with the project by using the hypertext call to the translator or by using the Google API through the program. Google also provides API for transliteration services.

5. QUERY EXPANSION, SEMANTICS AND DISAMBIGUATION

WordNet library has been used for collecting the related words of the query. These words are added to the query performing the query expansion and providing semantic to the search. WordNet is an online lexical database designed for use under program control. It is encoded in computerreadable form, and combines the features of a dictionary and a thesaurus. This makes it a valuable resource in natural language understanding, by providing the computer information that until now was not directly available. The lexical database contains English nouns, verbs, adjectives, and adverbs (open-class words).

WordNet® is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptualsemantic and lexical relations (Christiane 1998, George 1995, Princeton University).

The main relation among words in WordNet is synonymy, as between the words shut and close or car and automobile. Synonyms—words that denote the same concept and are interchangeable in many contexts—are grouped into unordered sets (synsets). Each of WordNet’s 117 000 synsets is linked to other synsets by means of a small number of “conceptual relations.” Additionally, a synset contains a brief definition (“gloss”) and, in most cases, one or more short sentences illustrating the use of the synset members. Word forms with several distinct meanings are represented in as many distinct synsets. Thus, each form-meaning pair in WordNet is unique.

The most frequently encoded relation among synsets is the super-subordinate relation (also called hyperonymy, hyponymy or ISA relation). It links more general synsets like {furniture, piece_of_furniture} to increasingly specific ones like {bed} and {bunkbed}. Thus, WordNet states that the category furniture

includes bed, which in turn includes bunkbed; conversely, concepts like bed and bunkbed make up the category furniture. All noun hierarchies ultimately go up the root node {entity}.

Hyponymy relation is transitive: if an armchair is a kind of chair, and if a chair is a kind of furniture, then an armchair is a kind of furniture. WordNet distinguishes among Types (common nouns) and Instances (specific persons, countries and geographic entities). Thus, armchair is a type of chair, Barack Obama is an instance of a president. Instances are always leaf (terminal) nodes in their hierarchies.

Meronymy, the part-whole relation holds between synsets like {chair} and {back, backrest}, {seat} and {leg}. Parts are inherited from their superordinates: if a chair has legs, then an armchair has legs as well. Parts are not inherited "upward" as they may be characteristic only of specific kinds of things rather than the class as a whole: chairs and kinds of chairs have legs, but not all kinds of furniture have legs. Verb synsets are arranged into hierarchies as well; verbs towards the bottom of the trees (troponyms) express increasingly specific manners characterizing an event, as in {communicate}-{talk}-{whisper}. The specific manner expressed depends on the semantic field; volume (as in the example above) is just one dimension along which verbs can be elaborated. Others are speed (move-jog-run) or intensity of emotion (like-love-idolize). Verbs describing events that necessarily and unidirectionally entail one another are linked: {buy}-{pay}, {succeed}-{try}, {show}-{see}, etc.

Adjectives are organized in terms of antonymy. Pairs of "direct" antonyms like wet-dry and young-old reflect the strong semantic contract of their members. Each of these polar adjectives in turn is linked to a number of "semantically similar" ones: dry is linked to parched, arid, dessicated and bone-dry and wet to soggy, waterlogged, etc. Semantically similar adjectives are "indirect antonyms" of the contral member of the opposite pole. Relational adjectives ("pertainyms") point to the nouns they are derived from (criminal-crime). There are only few adverbs in WordNet (hardly, mostly, really, etc.) as the majority of English adverbs are straightforwardly derived from adjectives via morphological affixation (surprisingly, strangely, etc.).

The words are grouped into sets of synonyms (synsets), each representing a lexicalized concept. The synsets are linked through semantic relations. Each word can be a noun, a verb, an adjective or an adverb, or combinations of these, depending on its context. WordNet has six semantic relations that can occur between synsets. Table I below illustrates these relations:

Table I: Six Semantic Relations of Synsets

1. Synonymy:	is the relation of similarity between two words.
2. Antonymy:	is the relation of oppositeness of two words.
3. Hyponymy:	is the relation between a subordinate and a superior. (Reverse is **hypernoymy**).
4. Meronymy:	is the part-to-whole relation. (Reverse is **holonymy**).
5. Troponymy:	is a relation between a manner of doing an action and the action. (has corresponding relation **hyponymy** for nouns)
6. Entailment:	is an implication between two actions.

In order to understand what a user is searching for, word sense disambiguation is required. When a term is ambiguous, it can have several meanings. Disambiguation processes make use of other information present in a semantic analysis system and take into account the meanings of other words present in the sentence and in the rest of the text. The determination of every meaning influences the disambiguation of the others, until a situation of maximum plausibility and coherence is reached for the sentence. All the fundamental information for the disambiguation process is represented in the form of a semantic network, organized on a conceptual basis.

In a structure of this type, every lexical concept coincides therefore with a semantic network node and is linked to others by specific semantic relationships in a hierarchical and hereditary structure. In this way, each concept is enriched with the characteristics and meaning of the nearby nodes. The semantic relationships (links), which identify the semantic relationships between the synsets, are the order principals for the organization of the semantic network concepts.

6. CONCLUSIONS

This paper proposes a search engine that tries to cross the language barrier for information search on the web. Similar to a semantic search, a cross lingual semantic web search engine is to present search results of different language pages based on the queried words in the users' native language. To implement such a search engine, the author uses the technology of Cross Lingual Information Retrieval (CLIR). The paper presents an approach for machine translation using UNL as an interlingua that intermediates between different languages. The search engine has been successfully implemented. It returns those page results also which don't have directly the keywords but contains the synonyms or related words as of the query.

REFERENCES

Alansary, Sameh *et.al*. The Universal Networking Language in Action in English – Arabic Machine Translation.

Bhattacharyya, Pushpak (????). Multilingual Information Processing Through Universal Networking Language. Technical Report, Indian Institute of Technology, Bombay.

Christiane, Fellbaum (1998, ed.). *WordNet: An Electronic Lexical Database*. MA: MIT Press, Cambridge.

Ellen, S. and Para, S. (1997). Mining Structural Information on the Web. The Sixth International WWW Conference (WWW 97). Santa Clara, USA, April 7-11.

George, A. Miller (1995). WordNet: A Lexical Database for English. *Commu. ACM*, **38(11)**, 39-41.

Landauer Thomas Ka, Foltz, P.W. and Laham, D. (1998). An Introduction to Latent Semantic Analysis. *Technical Report Discourse Processes*, **25**, 259-284.

Lawrence, P., Sergey, B., Motwani, R. and Terry, W. (????). The PageRank Citation Ranking Bringing Order to the Web.

Lisa, B. and Bruce, C. (????) Phrasal Translation and Query Expansion Techniques for Cross Language Information Retrieval. Center for Intelligent Information Retrieval Computer Science Department, University of Massachusetts Amherst, MA 01003-4610, USA.

Mukerjee, Amitabha, *et.al*. (2003). Universal Networking Language – A Tool for Language – Independent Semantics? International Conference on the Convergence of Knowledge, Culture Language and Information Technology, Dec. 2-6. Alexander Egypt.

Nawab, Md. Yousuf Ali *et al.* (2011). Conversion of Bangla Sentence into Universal Networking Language Expression. *Inter. Jour. Comp. Sci. Issues*, **8(2)**.

Nguyen Dat, P.T. and Ishizuka Mitsuru . A Statistical Approach for Universal Networking Language – Based Relation Extraction.

Philipp, S. and Philipp, C.(????) Cross- lingual Information Retrieval with Explicit Semantic Analysis. Institute AIFB, University of Karlsruhe.

Princeton University "About WordNet." WordNet. Princeton University. (2010). <http://wordnet.princeton.edu>

Ramanathan, A. State of the Art in Cross-Lingual Information Retrieval. *National Centre Software Tech.*, **140**, 26-41.

Sergey, B. and Lawrence, P. The Anatomy of a Large- Scale Hyper textual Web Search Engine. Computer Science Department, Stanford University, Stanford, CA 94305.