CHAPTER 2
LITERATURE SURVEY

2.1 PREAMBLE

This chapter presents an overview of the existing literature on the research work. Semantic search is the field of information retrieval system, which is distinct from the traditional information retrieval methods. The semantic search purpose is to help users to denote their search intentions and assist search engines to understand the meaning of users’ queries in terms of semantic web technologies. This concept understands from a broader sense, in that semantic search brings up to using semantic technologies for information retrieval. Figure 2.1 shows an extensive literature survey of IR and it was done by referencing various journals, conference proceedings and book volumes.

Figure 2.1 Literature survey of information retrieval
This chapter is structured as follows: section 2.2 presents information retrieval models, section 2.3 discusses the existing literature in the field of semantic search and technologies, section 2.4 provides common issues of the existing literature of information retrieval models and semantic search technologies, finally summary of the chapter is depicted in section 2.5.

2.2 INFORMATION RETRIEVAL MODELS

2.2.1 Set Theoretic Model

The CBR algorithm is based on set theoretic model. It is used to retrieve and reuse the existing problem solutions for emerging problems, which has four, sub-processes as follows (Cassidy et al 2003):

- **Retrieve**: a new problem is compared with cases in the database.
- **Reuse**: if there are be fitted to the cases, the results to the retrieved cases are reused as the results to the emerging problem.
- **Revise**: if the retrieved cases cannot entirely fit the problems, the results to the problem need to be revised.
- **Retain**: the new case, incorporating both problems and solutions, is stored in the database.

Every feature mined from incident reports is awarded an equal weight. Each feature in a new occurrence is matched with the corresponding feature in every of the other incidents. If the features match, a score of 1 is awarded. If the features do not match, a score of 0 is awarded. A similarity score is calculated by:

1. Finding the sum of the matching features; and
2. Dividing this sum by the number of features contained in the incident, given the formula below:

\[
sim(T, S) = \frac{\sum_{i=1}^{n} f(T_i, S_i)}{n}\quad (2.1)
\]
where $T$ is a new incident, $S$ is an existing incident, $T_i$ is a feature of incident $T$, $S_i$ is a feature of incident $S$, $n$ is the number of the features in the incident $T$ and $S$, $f_i \in \{0, 1\}$ is a function to match between $T_i$ and $S_i$. Then a threshold value is set up to determine whether or not the two incidents are matched (Dong et al 2008b).

The CBR model first entering commerce in the years 90s, it is applied in several domains. Bichindaritz and Marling (2006) used CBR model in health domain, particularly in diagnosis systems and biomedicine. Rissland et al (2005) proposed case based reasoning in law, as law generally depends on cases, it is an interesting domain for CBR researchers. Tseng et al (2005) applied case-based reasoning for product configuration in mass customization environments to reduce the prize of a new product, the algorithm CBR was used to build the new material (equipment) invoice. Getting back to cases which look like the present problem can create a big time saving during the elaboration of problem and a correct management to designers.

Dong et al (2010) proposed a conceptual framework of an ontology-based semantic service search engine for transport domain. This work built an Extended Case-Based Reasoning (ECBR) model for similarity matching based on CBR model. The model is to calculate the similarity value of a concept $c_j$ to a metadata $m$, which is represented as

$$sim(cd_j, sd) = \frac{\sum_{i=1}^{m} f(cd_k, sd)}{m}$$  \hspace{1cm} (2.2)

$$f(cd_k, sd) = \begin{cases} 
1 & \text{if } \exists sd_k \left( \forall k_i, g_i(cd_k) = g_i(sd_k) \right) \\
0 & \text{otherwise}
\end{cases}$$  \hspace{1cm} (2.3)

where $cd_j$ is the content of the definition of a concept, $k_i$ is an index term; $cd_j = (cd_{k1}, cd_{k2}, ..., cd_{km})$, where $cd_k$ is the index terms that occurs with $cd_j$, $m$ is the number of index terms that occur with $cd_j$; $sd$ is the content of the service
description property regarding a service metadata; \( sd = (sd_{k1}, sd_{k2}, ..., sd_{kn}) \), where \( sd_k \) is the index terms that occur with \( sd \), \( n \) is the number of index terms that occur with \( sd \); \( g_i \) is a function that returns a weight associated with \( k_i \). In this method low precision and recall rate that appeared in this experiment.

The CBR algorithm is simple and easy to implement. The returned similarity value is not a binary value but a decimal fraction between 0 and 1, which can be utilized in the ranking. However, the weight of features should be different according to their importance to belonged incidents. Moreover, it is difficult to take into account semantic matches by this algorithm (Dong et al 2008b).

### 2.2.2 Algebraic Model

The vector space model (VSM) is an algebraic model for representing text documents as vectors for information retrieval (Pablo et al 2007). Each document is represented as a vector, and each dimension of the vector corresponds to a term in indexed terms (Salton 1971). If a term appears in a document, a weight is assigned to a corresponding dimension in the vector. Similarly a query also can be seen as a vector with corresponding indexed terms. Thus, the relevance between a query and a document can be calculated as the cosine of the angle between the two vectors. A threshold is set up to determine the similarity between the terms from a document and a query, which enables a document to become partially similar to a query. The weight can be calculated in terms of term frequency, inverse document frequency (tf-idf) weight (Salton & Michael 1983). Term frequency is the raw frequency of a term in a document, which is often normalized to prevent a bias towards longer documents. Inverse document frequency is the inverse of the frequency of a term among the data. That is,
$sim(d_j,q) = \frac{|\overrightarrow{d_j} \cdot \overrightarrow{q}|}{|\overrightarrow{d_j}| \times |\overrightarrow{q}|} = \frac{\sum_{i=1}^{t} w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^{t} w_{ij}^2} \times \sqrt{\sum_{i=1}^{t} w_{iq}^2}}$ \hspace{1cm} (2.4)

where $\overrightarrow{d_j}$ and $\overrightarrow{q}$ are two vectors corresponding to document $d_j$ and query $q$, respectively. $|\overrightarrow{c_j}|$ and $|\overrightarrow{q}|$ are the norms of $\overrightarrow{c_j}$ and $\overrightarrow{q}$, $t$ is the number of terms in the terms list of the document, $w_{ij}$ and $w_{iq}$ are weights of each element of $\overrightarrow{c_j}$ and $\overrightarrow{q}$ corresponding to each term. Since $w_{ij} \geq 0, w_{iq} \geq 0, sim(c_j,q) \rightarrow [0,1]$.

The VSM, the tf-idf term weighting scheme improves retrieval performance. Its partially matching strategy can improve the recall of retrieved results. The cosine algorithm is to be used to rank and index the retrieved results. Nevertheless, the dynamical document bases make index terms difficult to maintain. In addition, the dependency of index terms is a prerequisite for VSM. Due to the locality of many term dependencies, the indiscriminate application to all documents in the collection might hurt the overall performance (Raj & Gopal 2012).

- **Latent Semantic Indexing (LSI) model**

LSI model is a type of algebraic model. The main idea of the LSI model is to map each document and query vector into a lower dimensional space associated with documents, which is used for information retrieval (Thorleuchter & Van 2013). LSI is to map each document and query vector into a lower dimensional space which is associated with documents. The documents are achieved by mapping the index terms vectors into this lower dimensional space. The LSI proposes to decompose a term-document association matrix in three components using singular value decomposition. The first one is the matrix of eigenvectors derived from the term-to term correlation matrix; the second one is the matrix of eigen vectors derived from the transpose of the document-to document matrix; the third one is a $r \times r$
diagonal matrix of singular values where $r$ is the minimum between the row and the column of the original matrix, and the rank of the term-document association matrix. Consider only the largest singular values of the third matrix are kept, along with their corresponding columns in the first and the third matrix while the rest singular values are deleted.

The resultant matrix is the matrix of rank $s$, which is closest to the original matrix in the least square sense. The relationship between two documents in the reduced space of dimensionalities can be obtained from the multiplication of the resultant matrix and its transpose. To rank documents with regards to a query, the query is modeled as a pseudo document in the original term-document matrix. Assume the query is modelled as the document with number 0. Then the first row in the multiplication of the resultant matrix and its transpose provides the ranks of all documents with respect to this query. Based on the index term list, each concept $c$ is formed as an array in which each element is obtained by tf-idf, and all the concepts in the ontology are formed as a term-concept matrix $A$ (Dong et al 2010). The term-concept matrix is then decomposed by the SVD (Singular Value Decomposition) approach, which can be mathematically denoted as Equation (2.5)

$$A = UV^T$$

where $U$ is the matrix derived from the term-to-term matrix given by $AA^T$, $V^T$ is the matrix derived from the transpose of the concept to concept matrix given by $A^TA$, and $\Sigma$ is a $r \times r$ diagonal matrix of singular values where $r = \min(t, N)$ is the rank of $A$. Considering that now only $k$ largest singular values of $\Sigma$ are kept along with their corresponding columns in $U$ and $V^T$, the resultant $A_k$ matrix is the matrix of rank $k$ which is closest to the original matrix $A$ in the least square sense. This matrix is given by Equation (2.6)
\[ A_k = U_k \Sigma_k V_k^T \] (2.6)

where \( k \) (\( k < r \)) is the dimensionality of a reduced concept space.

Analogous to the concept, a query \( q \) can be formed as an index term-based array in which each element is the tf-idf weight between the query and a term from the index term list. The array can then be translated into the concept space by Eq. (2.7), and then compared with \( A_k \) by the cosine algorithm to calculate the similarity values of each concept, which can be denoted by Eq. (2.8)

\[
q' = \Sigma_k^{-1} U_k^T q \tag{2.7}
\]

\[
sim(c, q') = \frac{|A_k \cap q'|}{|A_k| \times |q'|} \tag{2.8}
\]

LSI forms an efficient indexing scheme for the documents in the collection, and it supports for elimination of noise and removal of redundancy. Nevertheless, currently the LSI has not been validated on the large scale document retrieval system.

2.2.3 Probabilistic Model

The fundamental principle of a probabilistic model is to progress the probabilistic description of the ideal answer set of documents relating to a query by a series of interactions (Sparck et al 2000 & Wang et al 2010). The process of a query is as follows: Firstly, the user takes a look at the retrieved documents and initially guesses which one is related and which one is not, which groups a preliminary probabilistic description of the ideal answer set. Then the system can use this information to refine the description of the ideal answer set. By repeating this process many times, it is expected such a
description will evolve and become closer to the real description of the ideal answer set.

The guess of the fundamental of probabilistic model is given a user a query and a document in a collection; the probabilistic model tries to estimate the probability that the user will find the interesting document. The model assumes that this probability of relevance only depends on the query and the document representations only. Furthermore, the model assumes that there is a subset of all documents which the user prefers as the answer set for the query. Such an ideal answer set should maximize the overall probability of relevance to the user. Documents in the ideal set are predicted to be relevant to the query, and documents out of this set are predicted to be non-relevant.

The probabilistic model, documents are ranked in decreasing order of their probability of being relevant. However, in the implementation of probabilistic model, users could make mistakes in initially guessing the initial separation of documents into relevant and non-relevant sets. Additionally, the method does not take into account the frequency of index terms in a document, which cannot weight the importance of different index terms towards a document. Finally the assumption that all index terms are independent is not feasible in practice (Dong et al 2008b).

- **Bayesian Network**

  A Bayesian network is a type of probabilistic model. It is probability casual network for substituting causal relationship between variable. It is also known as a Bays net, belief network and which can be used for prediction, sensor fusion, classification, risk analysis, diagnosis, decision analysis, combining uncertain information and numerous probabilistic inference tasks. It consists of a set of nodes and directed arcs. The nodes depict variables and the arc represents the directed causal
influences among linked nodes. The arc is created from the parent node \((V)\) to the child node \((U)\). The child node is dependent on its parent node, but the child node is conditionally separate of others. The probability \(P\) \((V|U)\), showing how a given parent node, \(V\), can influence the probability distribution over its child node, \(U\), is calculated using Bayes’ Theorem (2.9):

\[
P(V/U) = \frac{P(U/V)P(V)}{P(U)}
\]

where, \(P(U|V)\) is the probability of \(U\) given \(V\), and \(P(U)\) and \(P(V)\) are the probabilities of nodes \(U\) and \(V\).

Bayesian network models were first introduced in IR by Turtle & Croft (1989). In their model, index terms, documents and user queries are seen as events and are represented as nodes in a Bayesian network. The model takes the viewpoint that the observation of a document induces belief on its set of index terms, and that specification of such terms induces belief in a user query or information need. This model was shown to perform better than traditional probabilistic models for the task of document ranking.

Second model was proposed by Ribeiro-Neto and Muntz (1996), where the elements of an IR system are formally defined as concepts in a sample space. Their work not only provides a probabilistic justification for the model, but also demonstrates that the combination of evidence from past queries with the vector space ranking yields better results than the use of a vector space ranking alone.

Acid et al (2003) presented a third model whose network topology is defined in such way that an exact propagation algorithm, proposed in their work, can be used to efficiently compute the relevant probabilities of the documents. When compared to Turtle and Croft’s work for the task of
document ranking, this model shows better performance in four out of five reference collections.

Bayesian networks have also been applied to other IR problems besides ranking as, for example, relevance feedback (Haines & Croft 1993), automatic construction of hypertext (Shin et al 1997), query processing (Ragunathan et al 2014), information filtering (Jabar et al 2014), document clustering (Buryak et al 2014), classification (Bu et al 2009) and tourism (Huang & Ban 2009, Hsu et al 2012).

### 2.2.4 Additional Models

In the above sections, set theoretic based CBR model, algebraic based vector space model (VSM) and latent semantic indexing (LSI), probabilistic model and Bayesian network model were presented. In addition, the fuzzy, Boolean and the extended Boolean models, the generalized vector model, the neural network models, inference network model, etc., an introduction of these models can be found in (Baeza-Yates & Ribeiro Neto, 2011).

### 2.3 SURVEY ON SEMANTIC SEARCH AND TECHNOLOGIES

Today, a number of semantic search technologies have been proposed and their application areas are diverse. This survey mainly focuses on six application areas of semantic search technologies that relate to our research, namely: XML based search technologies, ontology based search technologies, semantic search technologies, hybrid semantic search technologies, multimedia semantic search technologies, and natural language search technologies.
2.3.1 Semantic Web Based Search Technologies

Semantic search aims to capture data relationships and make resulting data queryable (Dong et al 2008a).

Sheth et al (2002) proposed a semantic content and retrieval engine (SCORE). Two types of metadata are extracted from documents, which are syntactic metadata describing non-contextual information about documents and semantic metadata describing domain specific information about documents. Based on the semantic metadata, documents are associated with context ontologies by means of automatic classification technologies. Users send queries to the system by specifying contexts and attribute values of metadata. This approach has the following limitations: The keyword-based matching style could produce incomplete results. The performance of the approach relies on the quality of ontologies and queries. The approach could be challenged by large-scale knowledge bases. They do not propose any means for ranking retrieved results. They do not propose any means for evolving the domain ontologies to make them consistent with the dynamic domain knowledge. They do not propose any means for dealing with the perceptual differences between ontology designers and users toward the ontologies.

Duke et al (2007) proposed a semantic search and browse tool Squirrel. By means of this system, once a user submits a query, the query terms will be matched with concepts from proton lightweight general purpose ontology. The query terms and matched topics are sent to web search engines. The documents that match the query terms and topics are classified according to their types. Under each topic, documents are ranked following the combination of the degree of relevance to query terms and the level of interest to user profile. The limitations of the approach are as follows: The
performance of this approach relies on the quality of ontologies and queries. The expanded query terms could burden web search engines and cause query flooding. The keyword-based query term-ontological concept matching could lead to a mismatch or incomplete matching. User profile-based document ranking cannot work when users search for documents with new topics. They do not propose any means for ontology evolution.

Many semantic web research works are carried in tourism field. Jakkilinki et al (2007) proposed an ontology based e-Tourism Planner that enables users to create an itinerary in one single application with this intelligent tool that builds on semantic web technologies. Kanellopoulos (2008) presented semantic based information for tourism destinations by combining the Peer2Peer paradigm with semantic web technologies. In the specific field of using semantics to provide better information for tourists there have been relevant and recent efforts in the literature. Pejic et al (2009) presents an expert system for tourist framework. The frame work discusses about developing a web application integrated with Google Maps in AJAX. Khorasani et al (2010) tries to elaborate the concept of intelligent tourism electronic portal and framework.

Schreiber et al (2008) proposed Multimedia E-Culture project as one of the semantic search systems demonstrates how the novel Semantic Web and presentation technologies can be deployed to provide better indexing and search support within large virtual collections of cultural heritage resources. To search, semantic paths, this system checks all RDF literals in the repository for matches to the given keyword and traverses the RDF graph until a resource of interest is established. To end, the results are clustered based on the paths from the matching literals to their result. This research has some similarity with other semantic approach, but it lacks the ability to assign weights to
properties and resources and the ability to identify the search completion threshold. The inadequacies are the most important problems with this system, for the reason they are two of the factors for expanding the semantic network. Moreover, information travels in one direction only in this limited system: always from the query object in the triple to the corresponding subject.

Saleh et al (2009) presented semantic annotation tool called AraTation for Arabic news on the Web. The tool is capable of extracting news named entities. Location ontology is used for that tool. The reported performance of the tool achieves an average precision of 67% and recall of 82% on a set of ten locations over 25 Web documents.

Bo et al (2010) proposed semantic retrieval model which consists of semantic annotation, resource collection, query analysis and results ranking. Domain ontology and two bilingual dictionaries are used to extract keywords for semantic annotation. A method which combines lexical relationship and semantic relationship is presented to analyze user's query. A modulate method is proposed to rank results. The authors illustrated that the preliminary experimental results show the capability of the proposed model to boost the precision and recall rates of webpage searching.

Fernandez et al (2011) has developed a complete semantic retrieval approach to cover the entire IR process, from a NL query to a ranked set of documents. Currently, techniques for content description and query processing in IR are based on keywords, so that it provides limited capabilities to capture the conceptualizations associated with user needs besides documents. The conceptual search aims to resolve the drawbacks of keyword-based models and identify with search meanings rather than literal strings, has been the effort of a wide-ranging of research in the IR system. In this research the vector space model is combined with the ontology-based information retrieval
to retrieve the relevant documents. Thus, they do not fully exploit the information indexed and functionalities provided by customary, large-scale web search engines.

Sridevi & Nagaveni (2011) presented a semantic similarity measure based clustering algorithm. The process performed document annotation followed by the clustering based on particle swarm optimization clustering algorithm. Paiva et al (2012) proposed a generic semantic search platform and defined four parts: concept definition, data insertion, data expansion and search process. The prototype uses the case study of a document search system, concretely the quality management system. In this approach experimental evaluation is inadequate.

Du & Hai (2013) discussed the approach based on semantic ranking of web pages. The work started with the analysis of intension and extension similarity that noted down the usage pattern and their hyperlinks. The semantic ranking given in the paper is stated in three aspects: (1) Hyperlink-based ranking, (2) Content based ranking and (3) Hyperlink-content-based ranking.

Blanco et al (2013) proposed repeatable and reliable semantic search evaluation. The methodology has been tested only with entity search and entity list search and not cover semantic search challenge of other retrieval scenarios such as search for consolidated objects (data integration and search), search for documents with embedded RDF (semantic document retrieval) and search for relations between objects relational search.

Uddin et al (2013) described the semantic similarity measures used for enhancing the process of IR in folksonomies. In the work annotation process of various web resources are based on the set of tags for effective
goals. Tag-user relationship, tag-resource relationship and WordNet were computed designed for semantic similarity. In this paper, the relationships between tag-user, tag-resource, and tag-tag co-occurrences were used to select semantically relevant tags. Jindal et al (2014) reviewed of ranking approaches for semantic search on Web.

2.3.2 XML Based Search Technologies

A number of studies have discussed semantics in XML keyword search. Most of them focus on how to effectively connect matches of keywords in a meaningful way.

Guo et al (2003) proposed XRank that connects keyword matches by the LCA (Lowest Common Ancestor) nodes that contain at least one occurrence of all keywords after excluding the occurrences of keywords in sub-elements that already contain all keywords. Cohen et al (2003) introduced XSEarch the concept of interconnection. Two matches are interconnected if the paths from these two nodes and to their LCA may not contain distinct nodes with the same labels except for themselves. A similar concept has also been proposed in (Guoliang et al 2007).

Zhiyuan et al (2007) proposed an XML indexing approach which allows every fully specified XML query to be answerable using a single lookup. These properties are then only used if they are found to be highly selective for a particular query and document collection.

Elghandour et al (2008) proposed a system that helps users decide which indexes to build for an XML document collection. The system is tightly coupled to the query optimizer using database statistics to perform cost estimation.
Pu & Yu (2008) discusses the problem of query cleaning (e.g. segmentations and correction) for keyword queries in a relational database context, but their aim is to improve the quality of keyword queries, and they do not consider the characteristics of XML data.

Zima & Jezek (2010) provided information on how web documents written in the XML language were rewritten into logic forms in terms of programs in Prolog. The XML language constituted the basis of many semantic web languages, and information in XML documents was usually retrieved with the help of procedural language called XQuery. Retrieving based on logic formulas gave the chance to take advantage of logical deduction and in this way to gain new originally hidden information.

Chang & Hwang (2011) focused on the development of a processing model for efficiently querying encrypted XML documents using XQuery. This model requires certain documents for efficient querying, that specifies how to encrypt the XML documents and the XML Schema of the original XML documents. This model allows for the efficient querying of encrypted XML documents, in terms of the computation required for decryption during the query process.

Termehchy & Winslett (2011) proposed Coherency Ranking (CR), a domain and database design independent ranking method for XML keyword queries that is based on an extension of the concepts of data dependencies and related information. Using coherency ranking, the outcomes of a keyword query are invariant under a class of equivalency-preserving schema reorganizations.

Liu et al (2013) analyzed the structure in a keyword query to gain a deep understanding of keyword queries. The XML keyword query has
structure, and exploits some key observations to characterize the query structure. Then explore the use of query structure to improve search results. The basic idea is that a result should match not only the terms in the query, but also the query structure. This approach discusses how to process keyword queries on XML with query structure awareness.

2.3.3 Ontology Based Search Technologies

Ontology-based search technologies refer to the semantic search and technologies designed to denote users’ query intentions by means of ontology concepts. Ontology-based information retrieval systems are discussed in many studies from different aspects.

Paralic & Kostial (2003) proposed an ontology-based approach to information retrieval where resources (i.e. documents) are associated with concepts in ontology. The concepts in a query are matched to the corresponding concepts in an existing ontology. Then the query concepts are matched with the document concepts and thus the matched documents are retrieved. Finally, the total similarity score is calculated. When compared to the vector model, Term Frequency and Inverse Document Frequency (TF-IDF) and the Latent Semantic Indexing (LSI) approach, their ontology based approach performed significantly better.

Alani et al (2003) proposed Artequakt project for automatic ontology based knowledge extraction from web documents. The Artequakt project links a knowledge extraction tool with ontology to achieve continuous knowledge support and guide information extraction. The extraction tool searches online documents and extracts knowledge that matches the given classification structure. It provides this knowledge in a machine-readable format that will be automatically maintained in a knowledge base (KB).
Corby et al (2004) presented Corese system, an ontology based search engine for the Semantic Web, which retrieves Web resources annotated in RDF(S) by using a query language based on RDF(S). Corese is able to approximately search the Semantic Web. Approximation is provided by employing inference rules and by computing the semantic distance of classes or properties in the ontology hierarchies. Specially, Cores retrieves Web resources whose annotations are specializations of the request, also it get back those resources, whose annotations refer to concepts and relations that are hierarchically close enough to those of the query.

Vallet et al (2005) have proposed the vector-based ranking model that takes advantage of semantic knowledge representation. Ontology-based information retrieval matches the relevance of a user generated query against a Knowledge Base (KB). The ontological information retrieval utilizes the relations between the keywords. This method inherits all the well-known problems of building and sharing well-defined ontologies, populating knowledge bases, and mapping keywords to concepts. In this approach, the keyword indices are replaced by an ontology KB. The semi-automatic document annotation and weighting procedure is the equivalent of the keyword extraction and indexing process. The system was tested with a document base taken from an online newspaper archive. The experimental results showed the clear improvement with respect to keyword-based search.

Ding et al (2004) proposed SWOOGLE a crawler-based system for discovering, indexing, and querying RDF documents. SWOOGLE mainly provides a search for semantic web documents and terms (i.e., the URIs of classes and properties). It allows users to specify queries containing conditions on the document-level metadata (i.e., queries asking for documents having .rdf as the file extension), and it also allows users to search for semantic web documents using RDF/XML as the syntax language. Retrieved documents are
ranked according to a ranking algorithm measuring the documents’ importance on the Semantic Web. In contrast to SWSE (Hogan et al 2011), which is mainly concerned with entity search over instance data, Swoogle is mainly concerned with more traditional document-search over ontologies. The challenge of their approach is: (i) The large-scale semantic networks may make it difficult to improve the efficiency of this approach; (ii) The performance of the search approach relies on the quality of queries. (iii) Incompleteness of search result is another issue because the search could miss some results when searching in a huge semantic tree, (iv) Halting problem – all relations in semantic have the feather of uncertainty, which means that some statements returned from semantic search cannot be proven by logic in practice.

Pablo et al (2007) presented an information retrieval framework using ontology in order to improve the accuracy. Basically, documents are annotated and ontology is constructed based on the annotations. Then, the annotations for each document are weighted by an adaptation of the $tf*idf$ measure. For a user request, the queries are retrieved from the ontology if the annotations are matched at that time the documents containing the annotations are selected as the query results. The result and documents are ranked by an adapted vector-space model which assigns higher scores to the documents containing maximum weighted annotations. The relationship between the annotations was used to find the relevant document through the ontology query processing; on the other hand the differences between the weights of the relationships were not considered.

Wang et al (2010) employed probabilistic topic models to learn relationships between a set of concepts. Particularly, they were employed to learn concept hierarchies. Kayed et al (2010) described that ontologies can find concepts to aid machines to deal with data meaningfully. Two steps
performed to find concepts, the first step involves developing the ontology concepts and the second step involves measuring the relevance-based on the ontology concepts. Re-ranking has also been accomplished for enhancing the relevancy rate of the results.

Wimalasuriya and Dou (2010) have tried to provide a clear review for the field of ontology-based information extraction. The domain ontology is used to construct the semantic representation from unstructured design documents. The concept graphs are built to recognize concepts and relationships. The results demonstrate that this proposed method performs better than the keyword-based search techniques.

Kara et al (2012) presented ontology based retrieval framework using semantic indexing and its application is soccer domain, which includes all the aspects of semantic web like ontology developing, data extraction, ontology population, inference, semantic rules, semantic indexing and retrieval. When these technologies are combined with the comfort of keyword-based search interface, obtain a user friendly, high performance and scalable semantic retrieval system. The evaluation results easily outperform both the traditional approach and the query expansion methods. The performance will be improved by implementing a word disambiguation module for lexical ambiguities and a mechanism that expands the index automatically according to the user feedback.

2.3.4 Hybrid Semantic Search Technologies

Selections of search methods are employed with semantic web technology and traditional method to improve the precision of text search.

Rocha et al (2004) proposed a hybrid approach for searching in the semantic web may meet the following challenges: The approach could be
challenged by large-scale semantic networks. The performance of the search approach relies on the quality of queries. The Spread activation algorithm has the issue of query incompleteness and query flooding. There is no semantic interpretation of the activation value flowing through the network. Some inferences in the graph are uncertain, which means some statements returned from semantic search cannot be proven by logic in practice.

Han & Chen (2006) presented a hybrid web search methodology – HWS, combining traditional text search and semantic search, to improve the performance of traditional search engines. Three algorithms are adopted in the search engine, first algorithm is used to mine associations from existing user profile ontologies; second algorithm is used to construct class hierarchies by means of a hierarchy clustering method; and third algorithm merge classes into class hierarchies. A ranking algorithm utilized in HWS concerns all entities and relations, and contextual similarities between two entities. The methodology has the following disadvantages: Class hierarchies cannot be modified by either users or designers, and so cannot adapt to changing knowledge in real environments and satisfy users’ requests for structure change. The semantic search is based on traditional keyword-based search, which has the issue of query flooding for each keyword in one query and thus could increase the cost of search time and resources, and increase the browsing time of users. The class hierarchies cannot represent more complicated relationships between entities in domain knowledge. The performance of the search approach relies on the quality of queries. The approach could be challenged by large numbers of documents.

Yoo (2012) presented a hybrid query processing method for personalized information retrieval on the Semantic Web. This work has suggested three requirements for personalized information retrieval on the Semantic Web. First, the contents on the Semantic Web should be expressed
in machine-readable format using ontologies. Second, the machine should be able
to assess the individual requirements of each user. Third, effective query
processing based on individual requirements should be executed with a
reasonable retrieval time. The drawback of this approach is the method uses a
limited set of objective or subjective terms.

Uzun et al (2013) presented a hybrid approach for extracting
informative content from web pages. This study proposes a two-step content
extraction approach. The first step consists of feature selection, traditional
document object model creation, and the machine learning method. This step
generates a well-formed document that contains rules used in the second step.
The second step extracts informative content efficiently using string manipulation
functions.

2.3.5 Multi-Media Semantic Search Technologies

Multimedia search and technologies refer to the semantic search and
technologies designed in the purpose of retrieving audios, videos (Tian et al
2012), and images (Wang et al 2012 & Tian et al 2010).

Multimedia searching has become an important research field,
particularly for understanding E-commerce and entertainment Web search.
However, there is a major need for new multimedia Web search techniques and
systems. People continually raise the bar of expectation for sophisticated Web
search engines. Due to the rapidly increasing usage of digital multimedia data, a
single piece of information can often be richly conveyed using multiple correlated
media, thus enabling users to concurrently receive information from multiple
sources (Tjondronegoro & Spink 2008).

Text is currently one of the most intuitive methods for searching
images since most of the available search engines still rely on text, and users
usually think of topics, keyword, or other high-level concepts in the form of text to write queries. Current Web search engines, such as Google Image (http://images.google.com), MSN image (http://search.msn.com/images), and Yahoo! Image (http://images.search.yahoo.com), have been fixed at the point of text analysis relevant to the multimedia material. However, the major limitation of this approach is the tedious and ineffective nature of manually annotating every temporal segments of a video or every region of interests in an image. Moreover, the process of automatic annotation by mapping low-level features into high-level semantic concepts is generally difficult as it needs machine learning and interpretations. For example, systems can use modelled domain knowledge to make sense and interpret the semantic meaning of video data by observing visual, audio and text features (Duan et al 2003).

Yoshitaka and Ichikawa (1999) surveyed query methods that utilize low-level features from image such as shape, spatial relation, color and texture; and video, including object motion, spatial-temporal relations. They have compared these features-based queries with semantic-based and knowledge-assisted retrievals including ‘query-by-subject’ (i.e. keyword) and ‘query-by-subject/object’ (i.e. derivation knowledge). The emerging applications and services from these types of retrievals include: finding images that are visually similar to a chosen picture or sketch; summarizing videos with thumbnails of key frames; finding video clips of a particular event, story, or somebody; and making a two-minute skim of an hour-long program (Chang et al 1998).

Kherfi et al (2004) provided a comprehensive discussion on the issues, techniques and systems of image retrieval from the Web and reviewed some prototypes. Lu et al (2005) presented Content-based audio retrieval (CBAR) leads to a more accurate classification than what can be achieved by content based image retrieval systems. For example, audio stream can be classified into music, speech, and noise, as well as some semantic details like cheering, applause, and
laughter. This raises a challenge for bridging the gaps between low-level features and high-level semantic.

Labra et al (2010) applied semantic web technologies and collaborative tagging to multimedia web information systems. The proposed system uses a search engine that combines both kinds of meta-data to locate the desired content of images and video. It will also provide browsing capabilities through the ontology concepts and the developed tags.

Richang et al (2013) presented a novel concept named Mediapedia which aims to construct multimedia encyclopedia by mining web knowledge. The Mediapedia distinguishes itself from traditional encyclopedia in its multimedia presentations and demonstration, complete automatic production, dynamic update and the flexible framework where each module is extensible to potential applications. This approach employed the affinity propagation algorithm in producing the exemplars from image pool, while using latent semantic analysis to associate exemplars to Wiki pages and utilizing document lattice model to perform Wiki pages summarization and assembled them for multimedia encyclopedia. This study can be deemed as an attempt at constructing Mediapedia by leveraging on web information the implementation results not as good as expected. This may be aroused by the assumption that the distribution of images from Flickr can automatically make a tradeoff between “typicality” and “diversity”. Conversely, this is not the truth for all the concepts. Enhancement can be created by taking into account the tags in producing the exemplar, and leveraging the images embedded in Wikipedia to facilitate better association and so on. An alternative approach to tackle the problem is to start this Mediapedia from Wikipedia, by identifying different senses of the concept by Wikipedia first and then associating them with images /audios.
2.3.6 Natural Language Search Technologies

Natural language search and technologies refer to the semantic search and technologies designed to seek answers for natural language questions by means of semantic technologies.

Lopez et al. (2011) developed a very recent system Power Aqua an ontology-based natural language interface (NLI) system which surpasses traditional systems by managing multiple ontology sources and high scalability. Since its NL processing module remains the same as in the previous AquaLog system. AquaLog is a portable natural language interface to knowledge base (NLIKBB) system which handles user queries in a natural language (English) and returns answers inferred from a knowledge base. The system uses GATE libraries (namely the tokenizer, the sentence splitter, the POS tagger) (Lopez et al 2007).

Cimiano et al. (2008) proposed ORAKEL as an ontology-based NLI system. It accepts English factoid questions and translates them into first-order logic forms. This conversion uses full syntax parsing and a compositional semantics approach. It can be ported into a different domain but such porting requires a domain expert to create a domain-dependent lexicon. The lexicon is used for an exact mapping from natural language constructs to ontology entities. A possible drawback of ORAKEL’s approach is that the system can neither handle ungrammatical questions nor deal with unknown words.

Damljanovic et al. (2010) presented natural language interface to ontologies combining syntactic analysis and ontology-based lookup through the user interaction. The first step in the SPARQL query generation is the transformation of the natural language query into a set of ontology concepts (classes, instances, properties, and literals), which is based on the assignment of a proper ontology concept to each word. If the system is not able to assign a
proper ontology concept to a word, then the user is called for selecting the correct one. The user selections are used for training the system in order to improve its performance. The second step is the construction of triples of ontology concepts, which are finally inserted into SELECT and WHERE clauses for generating a SPARQL query. The results of the evaluation of the obtained SPARQL query are shown to the user both in a tabular and in a graphical form.

Kaufmann et al (2007) designed the NLP-Reduce system which does not involve any advanced linguistic and semantic tools and depends on matching the query words to the KB instances. Its core part is a query generator which is responsible for creating SPARQL query given the words and the lexicon extracted from the KB. The major strength of the system is its good portability as it does not depend on any complex NLP query processing.

Ferrandez et al 2009 introduced ontology-based NLI system it covers a cinema/movie domain. Its target language is Spanish. it involves two components, the user query formulation database, and textual-entailment engine. Whereas the former component serves mainly for development and system training purposes, the latter is intended for an unknown query processing. The core of the system is a database of query formulations. The databank encloses a set of 54 clusters, each cluster represents one type of question and it has a representative query pattern which was derived from a set of training data. To each cluster is also concomitant with one SPARQL query but in the work evaluation, the system is not able to answer unknown ontology concepts and therefore it fails if the user poses a query using terms that are not present in the lexicon.

Habernal & Konopík (2013) developed semantic web search using natural language (SWSNL) system with accommodation, public transport, and
language domain. It covers the complete process which includes preprocessing, semantic analysis, interpretations, and implementing a SPARQL language to retrieve the results.

Sangers et al (2013) proposed semantic web service discovery using natural language processing techniques. The framework proposes a keyword-based discovery process for searching web services that are described using semantically enriched annotations done by means of semantic languages for service description. It makes an intensive use of natural language processing techniques and a WordNet-based similarity measure for matching words. The SWSD engine can search for web services based on user search keywords. A matching score is computed based on the similarity between the words in the user query and a Web service description. Experiments have been done to test the performance of three different matching algorithms. The Jaccard matching algorithm performs best for discovering exact matching web services, while matching using a similarity approach gives the best results for finding similar web services.

2.4 COMMON ISSUES OF INFORMATION RETRIEVAL MODELS AND SEMANTIC SEARCH

From this survey identify some common drawbacks of traditional information retrieval models and semantic search technologies shown as below;

- **Identify user’s perceptions**
  A user perception plays an important role in semantic search engines. The existing semantic search engines are not able to satisfy user’s requirements due to the difference between each person’s subjective perceptions regarding the objective world.
• **Ignorance of evolving knowledge structure**
  Knowledge structures in the semantic search engines cannot be frequently updated to suit the change of users’ requirements and the change of external environment.

• **Low precision and high recall**
  Some search engine cannot show their significant performance in improving precision and low recall.

• **Lack of semantics and lack of evaluations**
  Since many semantic search methodologies are only in the phase of conceptual model development, many of them have not been tested via the reasonable number of experiments.

• **Ambiguity/Polysemy:** A polysemous word has more than one meaning. When searching for documents with a word such as ‘‘play’’, related to a theatre piece, a search engine can return unrelated results such as a set of games for children.

• **Lack of synonym relations:** Words are synonymous if they have the same meaning. Words ‘‘irritated’’ and ‘‘annoyed’’ are very closely related; however, when searching for one of these words, found items will hardly contain the other word.

• **Lack of consensus:** The lack of consensus in the use of tags, especially as granularity is concerned, makes a traditional tagging system quite inefficient. To describe a particular item, different users may consider terms at different levels of generality/ specificity. For example, a user can tag a photograph as ‘‘bird’’, and another user can tag the same photo as ‘‘eagle’’.
2.5 SUMMARY OF THE CHAPTER

This chapter carried out an extensive literature survey of the existing literature on information retrieval and different application areas of semantic search. From the literature survey, it is concluded that the semantic web is an emerging research area. Numerous proposals are made regarding search engines, but they are executed to a specific goal. It is suggested to of its importance to develop a generic semantic search. It will be adapted to any system, any domain and any user. The next chapter introduces generic semantic search proposals. Moreover, user perceptions and time complexity is an important issue in semantic search. This issue is resolved in chapter 4. In addition, the above literature investigated many proposals lacking on evolving knowledge structure and experimental weakness is to be worked out in chapter 5. The above mentioned problems will be solved in the forthcoming chapters.