CHAPTER 1

INTRODUCTION

Information available on the web is more complicated and maintained in a critical manner. There are millions of heterogeneous Internet sites offering various services which make the Internet undeniable and unintelligible. With the explosive development and growth of information on the Internet and on social, business, government, research and commercial databases, it has become difficult for the user to analyze, track, summarize and extract the exact information from the Internet.

Traditional search engines are extremely good at document finding but they are limited to simple relevance-ranking queries and it cannot merge the information from multiple web pages. Existing searching techniques and retrieval methods provide conditional or limited assistance to the users in retrieving the exact information that they need.

Therefore, serviceability of the Internet depends to a large degree on the amount of automated, uniform, user friendly services, systems and tools. Since the maximum data on the web has been manipulated as textual data and each data has relationship with other, text mining placed an important role in the field of Information Extraction (IE). The data which is actually expected from the source is considered as important information (Jer Lang Hong 2010).
Data extraction is the process of accessing specific information from the web. It is easy to extract a direct data from the web. But extraction of data which required calculation and coordination of data from different webpages is complicated task. But we could not find any relation between web pages when we extract the data. It is very difficult to extract a sequence of more details from different web pages as a single result to satisfy the user’s need. To device the knowledge from the Internet or database, it needs a step by step filtered and a sequence of automated action for extracting perfect information from the web.

The extraction of sequence data are carried out by many request and response interaction over web and it needs large response time. We can simplify the information by automating a step by step extraction process using intelligent agent.

This research work is mainly related to text data extraction which is the major branch of data mining and it propose an approach for extracting single data as important information from different web pages for the purpose of improving accuracy and efficiency of actual data extraction. This approach mainly concentrates to find exact data which is processed or calculated or summarized from different web pages.

1.1 DATA MINING

Data mining is a powerful tool that can help to find patterns and relationships within the data. It can be defined as the process of automatically searching large stores of data to find patterns and trends that go beyond standard analysis. Data mining uses mathematical analysis to derive patterns and trends that exist in data. It is an interdisciplinary research field of data base systems, Statistics, Machine learning, Information Retrieval (IR) and gradual process consist of the following process.
The real world data may be incomplete, inconsistent. So the missing data will be filled or removed and each identified defects are treated by corresponding techniques. Data cleaning involves missing and redundant data in the source file. Data integration combines data from different sources. It inserts data into a single coherent memory data from different data source. In the data selection process, the relevant data is retrieved from the data sources. The data transformation process contains a basic data management tasks such as smoothing, aggregation, generalization, normalization and attributes construction. Sample analysis is the object of discovering interesting pattern in extracting a pattern set. Knowledge representation comprises of visualization techniques which are used to interpret discovered knowledge to the user.

Figure 1.1 shows that the different phases and the iterative nature of the data mining process. This does not stop when a particular solution is found. The result of data mining triggers new business question, which in turn can be used to develop more focused model. The different phases of data mining are listed here.

- Data Cleaning
- Data Selection
- Data Transformation
- Data Integration
- Data Mining
- Pattern Evaluation
- Knowledge presentation.
- Problem definition
- Data gathering and Preparation
- Model building and Evaluation
- Knowledge Deployment

Problem definition focuses on understanding the project objectives and requirements. Once the business problem is specified, it can be formulated as a data mining problem and develop a preliminary implementation plan. The data gathering and preparation phase involves data collection and exploration. The data quality and pattern of the data are identified in this phase. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order.

Figure 1.1 Data Mining process
Tasks include table, case, and attribute selection as well as data cleansing and transformation. In this phase, the information is improved which can be discovered through data mining. In model building and evaluation phase the various models can be applied to calibrate the parameter to the optimal values. If the algorithm requires data transformations, it steps back to the previous phase to implement them. The original business goal which is stated in first phase is evaluated in this phase. Knowledge deployment is the use of data mining within a target environment. In the deployment phase, insight and actionable information can be derived from data.

In data mining process, the data can be mined whether it is stored in flat files, spreadsheets, tables, or some other storage format. The important criterion for the data is not only storage format, it is applicability of the problem to be solved.

The key properties of data mining process are

- Automatic discovery of pattern
- Prediction of likely outcomes
- Creation of actionable information
- Focus on large dataset and databases.

1.2 TEXT MINING

1.2.1 Overview

The phrase “text mining” is generally used to denote any research area that analyzes large quantities of natural language text and detects lexical or linguistic usage patterns to extract nearly useful information. Text mining
is emerging in the web world because most of the data are available in the web pages are text data. It is the new research field in knowledge discovery. Mooney & Razvan (2007) defined text mining as the process of discovering new, previously unknown knowledge from unstructured or semi structured textual resources.

Till, text mining creates new relationships and hypotheses for experts to explore further. Several techniques have been designed for text mining including conceptual structure, association rule mining, episode rule mining, decision trees, and rule induction methods. The main applications of Information Retrieval are document matching, ranking and clustering.

1.2.2 Text Mining Process

Text mining process combines the following sequential tasks.

- Text preprocessing
- Text transformation
- Attribute selection
- Pattern discovery
- Interpretation or evaluation

Text preprocessing is the initial step of text mining. It includes text cleanup, tokenization, part of speech tagging, word sense disambiguation and semantic structures. In this step, the unwanted text will be removed from web page and the text converted from binary format will be normalized. Text transformation includes text representation and feature selection. Here, the text document is represented by the words it contains and their occurrences. Feature selection selects a subset of the features to represent a document. It can be viewed as creating an improved text representation. It involves stop
word removal and stemming process. Stop words are the most common words which are unlikely to help text mining. Stemming identifies a word by its root and reduces dimensionality.

Further level of dimensionality reduction is done in attribute selection process. It removes irrelevant attributes of text. At pattern discovery state, the text mining process merges with the traditional data mining process. Classic data mining techniques are used on the structured database that resulted from the previous stage. This is purely application – dependent stage. Finally the well suited result is found in the Interpretation stage or the results generated are used as part of the input for one or more earlier stages.

Information Extraction (IE) plays an important role in text mining as illustrated in Figure 1.2. Generally data mining assumes that the data could be mined is already is in the form of structured database.

![Figure 1.2 IE based text mining framework](image-url)
But unfortunately in many applications the data are available in the form of free natural language documents. Since IE addresses the problem of transforming a corpus of textual documents into a more structured database, the database constructed by an IE module can be provided to the Knowledge Discovery from Databases (KDD) module for further mining of knowledge. The related task of information extraction aims to find specific data in natural language text. The data to be extracted is typically given by a template which identifies a list of slots to be filled with substrings taken from the information. This template includes slots that are filled by strings taken directly from the document. Several slots may have multiple fillers for each domain. After constructing an IE system that extracts the desired set of slots for a given domain, a database can be constructed from a group of texts by applying the extraction patterns to each document to create a collection of structured records. Standard KDD techniques can then be applied to the resulting database to discover interesting relationships. Specifically, from the relationship the rules are induced for predicting each piece of information in each database field given all other information in a record.

1.3 INFORMATION RETRIEVAL

Traditionally, Informational Retrieval systems are not expected to return the actual information, but only document containing that information. It deals with the information relevant to the user’s query. A user who is in need of information formulates a request in the form of query written in a natural language. A retrieval system response by retrieving document that seems relevant to the query. This is an engineering account of the IR system. The retrieval is performed by matching the query representation with document representation. This can be depicted by the Figure 1.3.
The word term can be a single word or multiword phrases. For example, the sentence ‘Design feature of Information extraction system’ can be represented as follows.

\{design, features, information, extraction system\}

It can be expressed by the set of terms

\{design, features, Information, extraction system, information extraction system\}

These multi word terms can be obtained by looking at frequently appearing sequences of words, n-grams, part of speech tags or by applying Natural Language Processing (NLP) to identify meaningful phrases. In NLP, phrase normalization captures structural variation in phrases. For example, the three phrases text categorization, categorization of text and categorize text are normalized to give text ‘categorize’.

1.3.1 **Text Operations in IR**

The two most commonly used text operations are
- Stop word elimination
- Stemming

(i) **Stop word elimination**

The lexical processing of index terms involves elimination of stop words. Stop words are high frequency words which have little semantic weight and are thus unlikely to help in retrieval. Such words are commonly used in documents, regardless of topic and thus, have no logical specificity.

**Table 1.1 Sample stop word in English**

<table>
<thead>
<tr>
<th>about</th>
<th>above</th>
<th>accordingly</th>
<th>across</th>
<th>after</th>
</tr>
</thead>
<tbody>
<tr>
<td>afterwards</td>
<td>again</td>
<td>against</td>
<td>all</td>
<td>almost</td>
</tr>
<tr>
<td>alone</td>
<td>along</td>
<td>already</td>
<td>also</td>
<td>although</td>
</tr>
<tr>
<td>am</td>
<td>among</td>
<td>amongst</td>
<td>always</td>
<td>an</td>
</tr>
<tr>
<td>and</td>
<td>another</td>
<td>any</td>
<td>anybody</td>
<td>anyhow</td>
</tr>
<tr>
<td>anyone</td>
<td>anything</td>
<td>anywhere</td>
<td>apart</td>
<td>are</td>
</tr>
<tr>
<td>around</td>
<td>as</td>
<td>aside</td>
<td>at</td>
<td>away</td>
</tr>
<tr>
<td>awfully</td>
<td>be</td>
<td>because</td>
<td>been</td>
<td>before</td>
</tr>
<tr>
<td>besides</td>
<td>best</td>
<td>better</td>
<td>between</td>
<td>beyond</td>
</tr>
<tr>
<td>both</td>
<td>brief</td>
<td>but</td>
<td>by</td>
<td>can</td>
</tr>
<tr>
<td>could</td>
<td>did</td>
<td>during</td>
<td>each</td>
<td>even</td>
</tr>
<tr>
<td>even</td>
<td>ever</td>
<td>every</td>
<td>for</td>
<td>from</td>
</tr>
<tr>
<td>further</td>
<td>had</td>
<td>has</td>
<td>have</td>
<td>her</td>
</tr>
<tr>
<td>herself</td>
<td>him</td>
<td>himself</td>
<td>he</td>
<td>furthermore</td>
</tr>
<tr>
<td>many</td>
<td>near</td>
<td>shall</td>
<td>she</td>
<td>self</td>
</tr>
<tr>
<td>whose</td>
<td>why</td>
<td>will</td>
<td>where</td>
<td>ex</td>
</tr>
<tr>
<td>except</td>
<td>far</td>
<td>first</td>
<td>five</td>
<td>former</td>
</tr>
<tr>
<td>formerly</td>
<td>over</td>
<td>overall</td>
<td>usually</td>
<td>appropriate</td>
</tr>
</tbody>
</table>
(ii) Stemming

Stemming normalizes morphological variants by removing affixes from the words to reduce them to their stem. For example, the words compute, computing, computes and computer are all be reduced to same word stem, ‘compute’. Thus the stemmed representation of the text ‘design feature of information extraction system’ is

\{design, feature, inform, extract, system\}

It is very important to differentiate IE and IR. Information Extraction is quite different from Information Retrieval;

- An IR system finds relevant text and presents them to the user.
- An IE framework analyzes text, summarize it according to the pattern and present only the specific data from them that the user exactly needs.

1.4 INFORMATION EXTRACTION

1.4.1 Information Extraction and the World Wide Web

The World Wide Web (WWW) is nowadays referred as the unstructured or semi structured web of documents. Due to advancement of the web, digital libraries, technical document and medical data, the access to an immense amount of textual document become more and more effective. The knowledge discovery from textual database, text mining is an essential and difficult challenge due to the richness and ambiguity of natural language. So the necessity of structured text is getting increased in order to extract Information with pre-specified types of events, entities and relationship of user’s query.
Information Extraction is a process of indentifying actual data from large amount of data. It is very ancient technique initiated at the early days of 1970’s and now reached at multimedia document processing, automatic annotation and extraction of data on image, audio and video. Previous work in this scenario had attempted similar things in some cases, and indeed a related term “fact extraction” was in use as far back as the 1960s. Initially it was implemented for the purpose of providing real-time financial news to financial traders. It is a sub task of searching or Information Retrieval system. It can be used for indexing purpose of Information Retrieval (IR) system such as Internet search engine. Unlike Information Retrieval, Information Extraction produces structured data ready for further processing of text mining and its applications.

Figure 1.4 Web page shows the dengue fever details of Tirunelveli District
The above screenshot shows that the detail of the dengue fever spread over in Tamilnadu. Actually user only needs the information of number of person affected by dengue fever. This web page has the related additional information with the exact information. The actual extracted data is shown in a Table below.

<table>
<thead>
<tr>
<th>Disease</th>
<th>Location</th>
<th>No. of persons affected</th>
<th>Time period</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dengue fever</td>
<td>Tirunelveli</td>
<td>29</td>
<td>June 2012</td>
</tr>
</tbody>
</table>

1.4.2 The Tasks of Information Extraction

Traditionally information extraction tasks assume that the structures of the data to be extracted. For example, the types of named entities, the types of relations, or the template slots, are well defined. The main fundamental tasks of Information Extraction are

- Named Entity Recognition (NE)
- Co-reference resolution (CO)
- Template Element construction (TE)
- Template Relation construction (TR)
- Scenario Template production (ST)

(i) Named Entity Recognition

Named entity recognition is the most fundamental task in information extraction. It is a set of words that depicts some real world entity. It may be a name of the person, company name or the place. The task of
named entity recognition is to identify named entity from text and classify them into a set of predefined types like names, organizations and locations. Early solutions to named entity recognition rely on manually created pattern. It requires human expertise and labor intensive to create such patterns. It implements two approaches namely

- Rule based approach
- Statistical learning approach

**Rule Based Approach:** In named entity recognition, the rule based approach works as follows: It is a set of rules either manually created or automatically learned. Every token within the text is described by a group of options. The text is then compared against the foundations and a rule is fired if a match is found. A rule consists of a pattern and an action. A pattern is usually a regular expression defined over features of tokens. When this pattern matches a sequence of tokens, the specified action is fired. It is possible to match multiple rules for a sequence of tokens. To avoid such conflicts, a set of policies has to be defined to control how rules should be fired. One of the approaches is to order the rules in advance so that they are sequentially checked and fired.

**Statistical Learning Approach:** Most of the research work on named entity recognition is based on statistical machine learning. Many statistical learning-based named entity recognition techniques handle the task as a sequence labeling problem. Sequence labeling is a general machine learning method and has been used to model many natural language processing tasks including part of speech tagging, chunking and named entity
recognition. To map named entity recognition to a sequence labeling problem, each word in a sentence is treated as an observation. The class labels have to clearly define both the boundaries and the types of named entities within the sequence.

(ii) Co-reference Resolution

Co-reference resolution (CO) involves identifying identity relations between entities in texts. These entities are both those identified by NE recognition. For text searching purposes, Co-reference resolution might be useful to highlight all occurrences of the same object or provide hypertext links between them. This technology might also be used to make links between documents. The main significance of this task is to act as a building block for template element construction and scenario template production.

(iii) Template Element Construction

The template element construction task builds on Named Entity recognition and Co-reference resolution, associating descriptive information with the entities. The format of TE is arbitrary one. It may be a database record, a well formatted SQL operation, reading into a spreadsheet and multilingual presentation. As in named entity recognition, the productions of TEs are weakly domain dependent, since the change on any domain would involves some changes to the system.

(iv) Template Relation Construction

Initially relations between entities were part of the scenario and the specific template would be the output of IE evaluations. The template relation construction task requires the identification of a small number of possible relations between the template elements identified in the template element
task. This might be, for example, an employee relationship between a person and a company, a family relationship between two persons, or a subsidiary relationship between two companies. Extraction of relations among entities is a central feature of almost any information extraction task.

(v) Scenario Template Production

The end task in Information Extraction is Scenario template production. ST is the process of detecting and characterizing the semantic relations between entities in text. The two entities involved in a binary relation is referred as arguments. Feature based classification technique, kernel method and supervised learning method are the techniques involved in this task.

1.4.3 Applications of IE

Information extraction is useful in a different set of application which can be categorized as enterprise, personal, scientific and web oriented.

(i) Enterprise Application

**News Tracking**: It is a main application of information extraction which has covered a major research in the NLP domain. It automatically tracks the specific data from the news source. It mainly extracts the structured entities like people, company and relation among them. It also concentrates the tracking of disease details and terrorist information from the news source.
Figure 1.5 Applications of IE

Customer care: Any customer-oriented enterprise collects many forms of unstructured data from customer interaction. So that they have to maintain a structured database and business ontology to manage the data. Many extractions are involved in this application like product name identification, attributes of product, linking of customer requirement mail with the corresponding transaction in a database and the customer mood from phone conversation transcript.

Data Cleaning: Data cleaning is mainly used in maintaining large databases and data warehouses. Cleaning includes conversion of flat string details in to their structured form of the database.

Classified Ads: Classified advertisements and other listings such as restaurant lists is another domain with implicit structure that when exposed can be invaluable for querying. Information extraction applies such a kind of these record oriented data.
(ii) Personal Information Management (PIM)

PIM seeks to organize personal data like emails, projects and related people in a structured inter-linked format. It can automatically extract structure from existing file-based unstructured sources.

(iii) Scientific Application

Bio-informatics is the main field of extraction application which extracts from named entities to biological objects such as proteins and genes. Other areas like Nano science, Medical Instrumentations and chemical engineering applies extraction process for maintaining structured data of their field.

(iv) Web Oriented Applications

Citation Database: Many popular citation databases like Citeseer® and Google Scholar® are created through elaborate structure extraction steps from conference web sites to individual web pages. The creation of such databases requires structure extraction at many different levels starting from navigating web pages locating pages containing publication records, extracting title, author and reference from research paper. The resulting structured database provides significant value added in terms of allowing forward references and aggregate statistics such as author-level citation counts.

Opinion Database: There are many web sites storing immoderate opinion about music, movie, products and books. Many of the opinions are in free text form hidden behind blogs, newsgroup posts, review sites, and so on. The value of these reviews can be
greatly enhanced and organized along structured fields. It will be useful to find out exact feature of the product and the prevalent polarity of the opinion to buy the product.

**Comparison Shopping:** As web technologies evolved, nowadays most of the merchant web sites moved towards getting hidden behind forms and scripting languages. Consequently, it has been shifted to crawling and extracting information from form-based web sites. The extraction of information from form-based web sites is one of the important research areas. There exist quit interest in designing of comparison shopping websites that automatically crawl merchant websites to find products and their prices which can be used for comparison shopping.

**Advertisement Placement of web pages:** The extraction of mentions of selling products and the type of opinion expressed on the product on the advertisement page are the examples of Information extraction task that can lead the Internet Advertisement placement Industry.

**Web Searching:** Generally keyword search is adequate for getting data about entities, which are typically noun or noun phrases. They failed on queries that are expecting relation between entities. It is a big challenge for information extraction and is allowing structured search queries including entities and their relationship on the web.
1.5 INTELLIGENT AGENT

1.5.1 Overview

In the recent years, there are many research topics and open issues from either part of agent and data mining interaction. In particular, issues for agent-driven data mining, and issues for mining-driven agents are attracting research interest. These issues are significant because of their fundamental and necessary roles in establishing a symbiotic relation between agent and data mining. It creates a continuous growth of interest in the design of intelligent agent architectures for dynamic and unpredictable domains. Agent can be implemented in any ways like software, networks, people and machines.

An intelligent agent is one that is capable of flexible autonomous action in order to meet its design objectives. The main objectives are as follows.

**Reactivity**: Agents are able to perceive their environment, and respond in a timely fashion to changes that occur in order to satisfy its design objectives.

**Pro-activeness**: Intelligent agents are able to exhibit goal-directed behavior by taking the initiative in order to satisfy its design objectives.

**Social ability**: Intelligent agents are capable of interacting with other agents (and possibly humans) in order to satisfy its design objectives.

The agent takes sensory input from the environment, and generates output actions that affect it. The interaction between the agent and the
environment is usually an on going, non-terminating one. Figure 1.6 represents an abstract, top-level view of an agent.

In the Figure 1.6, the input is perceived by the detectors and transferred to parse input. It recognizes and normalizes the data and also it deals with inconsistencies of data. In the action determination state, the goal is examined from the plan. The next state determines the invoked mechanism from chosen action. The action output generated by the agent is affecting its environment. Normally, an agent will have a set of continuous actions available to it. This set of possible actions represents the agents to modify its environments.

In most domains of reasonable complexity, an agent will not have complete control over its environment. It will have at best partial control, in that it can influence it. An agent-based system might be conceptualized in terms of agents, but it may be implemented without any software structures corresponding to agents at all.

An agent is responsible for satisfying specific goals. There can be different types of goals for the agent such as achieving a specific status and maximizing a given function. Every Intelligent agent is working individually
and has communication and cooperation with other agent to achieve a common goal. Because of the limited knowledge resources, agents may have inadequate problem solving capabilities. They must have to learn how to respond exactly to unexpected tasks while simultaneously carrying out their pre-programmed function.

It has the following features

- **Reactive**: Responds to changes in the environment
- **Autonomous**: Control over its own actions
- **Goal-oriented**: Does not simply act in response to the environment
- **Temporally continuous**: It is a continuously running process
- **Communicative**: Communicate with other agents, perhaps including people
- **Learning**: Changes its behaviour based on its previous experience
- **Mobile**: Able to transport itself from one machine to another
- **Flexible**: Actions are not scripted
- **Character**: Believable personality and emotional state

As agent technology becomes more established, a variety of software tools become available for the design and construction of agent-based systems.
1.5.2 Agent Architectures

Agent architecture is a specific methodology which implies how the construction of an agent can be decomposed into the construction of a set of component modules and how these modules should be made to interact within agent and environment. There are three classes of architecture.

- Deliberative Architectures
- Reactive Architectures
- Hybrid Architectures

(i) Deliberative Architectures

This approach designs an agent as a type of knowledge-based system. It contains a symbolic model of the world in which decisions are made via logical reasoning based on pattern matching and symbolic representations. The architecture needs to translate the real world into a corresponding symbolic description and the agents need to reason with this data quickly enough for the results to be useful.

(ii) Reactive Architectures

In this architecture, intelligent agent behavior can be designed based on assumption without an explicit representation and abstract reasoning. This architecture identifies two key points.

Situatedness and embodiment: Real intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems.

Intelligence and emergence: Intelligent behavior arises as a result of an agent’s interaction with its environment.
(iii) Hybrid Architectures

Both the classical and the alternative approach to agent architecture have their own limitations, hence neither a completely deliberative nor completely reactive approach is suitable for building agents. So hybrid architecture is the combination of deliberate and reactive architecture. A deliberative architecture is containing a symbolic world model, which develops plans and makes decisions in the way proposed by mainstream symbolic Artificial Intelligence (AI). A reactive architecture is capable of reacting to events that occur in the environment without engaging in complex reasoning.

1.5.3 Agents versus Objects

Agent Oriented Programming (AOP) introduces a number of new concepts that are alien to Object Oriented Programming (OOP). It is considered as a next evolution of Object Oriented Programming. The main characteristics of agent oriented programming are situatedness, autonomy and flexibility.

The situatedness represents that agents receive input from an environment and perform actions that may change the environment. Autonomy represents the software system should operate without the direct intervention of human being or other agents and the agent act to the environment by its own. Agents must be flexible in the sense that they should be both reactive and proactive. Reactivity implies that agents must take timely actions in response to changes in the environment. Pro-activity indicates that agents not only react, but also exhibit goal-oriented behavior.
Object Oriented Approach

- OOP views the computational system as made up of modules (classes and objects).
- Each module carries out a particular task.
- Modules can communicate with each other and have individual ways of handling messages.

Agent Oriented Approach

- AOP specializes the framework by setting the state of the agents with main components such as beliefs, capabilities, and decisions.
- Various constraints are placed on the state of the agent.
- A computation consists of these agents informing, requesting, offering, accepting, rejecting, completing and assisting one another.

Difference between Agents and Objects

Even though both objects and agents communicate with other objects or agents through message passing, in case of objects, message from other object is simply a request to carry out some task. The object to which the request is sent knows how exactly the task is to be carried out. All the tasks which an object can perform are specific, in other words, objects can only perform tasks they are trained for. Agents, on the other hand can affect the behavior of other agent through message passing.
An agent may not react to method invocations by other agents. Such refusal behavior is not inherent in object-oriented systems. Agent-oriented systems do permit inheritance, but such inheritance is distinguished from inheritance in object-oriented systems. Agents may inherit plans or actions, beliefs and goals. Here plans or actions and beliefs are correspondingly similar to methods and instance variables in object oriented programming.

1.5.4  The Limitations of Agents

Although agent technology has an important role to play in the development of leading-edge computing applications, it has some limitations. It should be noted that the very nature of the agent paradigm leads to a number of limitations, common to all agent-based applications which are:

- No overall system controller
- No global perspective
- Trust and delegation

**No overall system controller:** An agent-based solution may not be appropriate for domains, in which global constraints have to be maintained.

**No global perspective:** The actions are determined by that agent’s local state. However, since in almost any agent based system, complete global knowledge is not a possible.

**Trust and delegation:** For individuals to be adoptable with the idea of delegating tasks to agents, they must first trust them. Both individuals and organizations will thus need to become more
accustomed and confident with the notion of autonomous software components, if they are to become widely used.

1.6 PROBLEM IDENTIFICATION

Extracting information about a specific topic in sequence is an essential thing to get the entire idea about that topic and the identification of right information becomes more difficult. To derive the knowledge from the Internet or database, it is essential for the users to use the automated and intelligent tools due to the explosive development of data source available on the Internet. There is a growing consensus in the Internet society that one of the most reliable solutions to the problem of Internet information retrieval is the use of software agent technology.

The objective of this work is to extract the documents from the Internet by giving a keyword to the search engine. Thereafter, the name entity recognition is used to categorize the locations, name of the persons, organizations, etc. After the name entity recognition, the required fields are extracted from the documents which are collected by giving a keyword to the search engine. The extracted fields are then stored in a database for further processing. The data transfer agent is used to transfer the data which we collected from the documents to the Extraction System. The Extraction System would perform the interval splitting, extracts the locative disease pattern, and extracts disease affecting sequence and disease spreading sequence. Using this technique, it is easy to identify the diseases which were affected in a particular location, the sequence of diseases affected in a particular place and spreading sequence of a particular disease.
1.7 PROPOSED CONTENT OF THE THESIS

The rest of the thesis work is organized as follows: the second chapter describes the detailed literature review of information extraction and information retrieval in Text mining. It also describes the involvement of agent in text mining process. The third chapter gives the detail about an intelligent agent approach for extracting a sequence of information by using the derived mining algorithm. Java Agent DEvelopment environment (JADE) and performance evaluation mining algorithm are discussed in fourth chapter. Followed by conclusion of this research work and future direction are discussed.