Chapter 6

IMPLEMENTATION AND RESULTS

6.1 INTRODUCTION:

The Publically indexable Web or Surface Web is open and freely accessible to search engines. The Hidden Web which is also a part of the World Wide Web is inaccessible to traditional search engine’s web crawling technology.

In order to extract and index the data from Hidden web sources, “Hidden Web Crawler Based Search Engine (HWCBSE)” has been developed and implemented in ASP.net. It works in four steps that are given below:

1. Hidden Web Crawling
2. Data Extraction
3. Repository Formation
4. Query processing

A detailed discussion on the implementation of all steps followed by HWCBSE is given in the following sections. The snapshots of the outputs of each step are also shown in the respective sections.

6.2 HIDDEN WEB CRAWLING:

To crawl the hidden web pages, a novel method “Hidden Web Crawling” has been developed in section 5.1. To find the Hidden Websites of “used car” domain, “used car” keyword is placed into search box of Google search engine and result page obtained thereafter is shown in figure 6.1. It may be noted that this result page contains the list of URLs of Hidden Websites. A predefined list has been made by picking 22 URLs from result page of Google search engine for implementing the crawling process.
Now, the URLs are taken one by one to download its web page. Since, search interfaces are the entry points to these sites, Hidden Web Crawler applies interface_detect2( ) (section 5.1, figure 5.7) algorithm to all 22 sites. It first detects the search query interfaces by looking at domain features (domain ontology table built in section 5.1) in source code of every page. Output of detection of interface_detect2( ) function is shown in figure 6.2. This module returns a valid alert if a particular web page contains search interface of used car domain and invalid for the other web pages (like registration forms, sign up forms or simple web pages). As soon as the interface is detected, the Hidden Web Crawler extracts the code that lies between <FORM> and </FORM> tag.
After extracting the interfaces, next step is to fill the search form of Hidden website with attribute values of mandatory field (“city” in this case). Since this crawling process is tedious, complicated and also needs large memory to store the data, only two values for the attribute “city” were submitted as shown in figure 6.3.
For testing purpose, two different Hidden Websites (structured i.e. www.autonagar.com and semi-structured i.e. www.carsingh.com) from list of valid websites were chosen. As soon as Hidden Web Crawler submits the search form of both Hidden websites (see figure 6.3), two different result pages in response to this query were returned as shown in figure 6.4 and figure 6.5.

Fig 6.3: Submission of attribute values (faridabad, delhi) to two Hidden websites
Fig 6.4: Result page of www.autonagar.com for attribute “city” and value “faridabad”

Fig 6.5: Result page of www.carsingh.com for attribute “city” and value “faridabad”
After retrieving the result pages, Crawler starts extracting the data records from these pages as discussed in next section.

### 6.3 DATA EXTRACTION:

Hidden Web Crawler crawls the structured website (www.autonagar.com) by calling function structured_dataextraction ( ) (section 5.7.5.2, figure 5.35) and results extracted thereof get stored in a table called “car” as shown in figure 6.6.

Figure 6.6 shows that results are accurately extracted from the result page shown in figure 6.4. Now, the second website (www.carsingh.com) with semi-structured presentation of data is selected and sent to data preprocessing system ( figure 5.39). The data records are then extracted and stored in second table named “carwale” as shown in figure 6.7.
Figure 6.7 again shows promising results. After extracting data from both sites accurately and storing them in the tables, next step is to form a main repository which contains all the data records of both the sites. This step is discussed in next section.

### 6.4 REPOSITORY FORMATION:

To merge the results from both the tables (figure 6.4, figure 6.5), algorithm repository_formation( ) (section 5.8, figure 5.42) is called by the crawler and results obtained thereafter are shown in figure 6.8.
After storing Hidden web data, HWCBSE is now ready to process a user query. The proposed query processing is discussed in next section.
6.5 QUERY PROCESSING:

HWCBSE provides a user friendly search interface as shown in figure 6.9 where user can enter his query in the form of keywords.

![Search Box of HWCBSE](image)

**Fig 6.9: Search Box of HWCBSE**

When user writes query (keyword or set of keywords) in the search box of HWCBSE, *Query Processor* processes the query by calling function Queryprocess( ) ( section 5.10, figure 5.47 ) to find the records that exactly match the query. A query “maruti 2004” is fired in search box of HWCBSE and the result is shown in figure 6.10.

![Query Result for query “maruti 2004”](image)

**Fig 6.10: Query Result for query “maruti 2004”**

The same query “maruti 2004” was fired at Google search engine and result page returned in response is shown in figure 6.11.
It may be noted that when the link from list of result indexes is opened as shown in series of figures 6.12 to figure 6.15, it is very clearly seen that no quality information is returned. The result pages are the documents where the keyword “maruti” and “2004” lies. Figure 6.12 shows the history and generations of Suzuki swift while figure 6.13, figure 6.14 show the year wise sale information of “maruti suzuki” cars. The fourth opened link contains some information in structured form but this information does not exactly matches with the given query. It shows only one result i.e “Maruti Baleno lxi 2004” instead of showing all cars of maruti 2004. In contrast to google results, the proposed system “HWCBSE” returned most relevant information and accurate data records from different websites in summarized form.
No relevant data record, only text

No data record
Fig 6.14: Result page after opening link

Fig 6.15: Result page after opening link
Now, a large number of queries were fired to check the accuracy of the proposed mechanism and the results obtained thereof are shown in a series of figures i.e figure 6.16 to figure 6.19.

Fig 6.16: Query Result for query “maruti”

Fig 6.17: Query Result for query “maruti faridabad”
HWCBSE also provides additional facility to user. If user finds any data record interesting then upon clicking the URL given in that record as shown in figure 6.20, it opens the result page of the site from which the data was extracted (shown in figure 6.21, figure 6.22). So, user can see additional information if he wants.
Fig 6.20: Query Result for query “maruti”

Fig 6.21: Web page opened on clicking hyperlink
Fig 6.22: Web page opened on clicking another hyperlink

It may be noted that the efficiency of proposed Hidden Web Search Engine “HWCBSE” in terms of relevancy and accuracy is higher as compared to traditional search engine.