Change Detection in Web Page and in Flat Files

Parul
Research scholar
Department of Computer Science & Applications
Kurukshetra University, Kurukshetra

Pardeep Kumar, PhD
Assistant Professor
Department of Computer Science & Applications
Kurukshetra University, Kurukshetra

Shalini Aggarwal
Teaching Fellowship
Department of Computer Science & Applications
Kurukshetra University, Kurukshetra

ABSTRACT

Users are interested in current web pages contents in optimum time. In this paper, we describe the technique to detect multiple changes in a web page or Flat files in form of insertion or deletion of the content or structure change. This efficient approach is suitable for client application and for implementing server applications that could serve the needs of users in monitoring modifications to HTML web pages or to flat files made over the time, and that allow reporting and visualizing changes in order to gain insight about the significance and types of such changes.

Keywords

HTML, Web Page Change Detection, flat files, htmldiff, change blindness, visual attention.

1. INTRODUCTION

A user often uses a search engine to find a page or document of interest and then the user bookmarks the page of interest and wishes to re-visit the page to obtain a fresh copy of the page when it has changed in an interesting way. Obviously the first problem is to know when a page has changed in any way that is interesting and in result of that, fresh copy should be retrieved. Furthermore, a user interested in monitoring a large number of web pages, and want to be notified each time when there is change in the page/s whether he revisit the page or not. Because of the large number of pages of interest, it will be difficult to remember the concrete details about every page. Thus, the second important problem is to identify what and where the page has changed, including the types and the amount of changes between the fresh copy and the copy last seen. It is shown that under a variety of conditions observers are often unable to see large changes directly in their field of view. A user of the web might identify the way a particular page has changed since it was last viewed [1] For example; consumers can use the technology to monitor new products, the services or auctions on e-commerce websites. In an online shopping website, if a user likes an item but the price of that item is quite high at that time so he just bookmarks the item to gain high profit on it. Each time when the price is changed user will get a notification, so that user can easily get the item at new price. The technique is used by flipkart, snapdeal and many other online shopping websites.

Several tools have become available to address the problem of determining when a page changes, what is the change by direct send notifications to user by personalized Email or text message. Now a day browsers have extension or plug-in which shows the web page change detection. The user can select specific area in the web page which is to be monitor. These extensions also enable desktop notifications. For example in google crome PageMonitor and in firefox AlertBox.

We have developed a system that efficiently tracks the page change, compactly stores versions on a per-user basis, compares and presents the differences between pages.

In 1996, Htmldiff named program was developed for display the difference between two web pages in SML of NJ. In this program LCS algorithm was used two strings comparison which is described by Hirschberg in 1977. This program is also used in UNIX-diff program.

The program had some shortcomings as follows:

a) Computational Expensive: - for comparing two versions of same file of 100 Kbyte each having 800 anchors and 4000 words, it takes 3 minutes on 200 MHz SUN Ultra1 system.

b) Aesthetical Display of difference: - the difference of two versions of a file is shown as a "merged" page, marking up the latest version in place with font changes to highlight the changes.

c) Not Platform Independent: - This version is only supported by the platform on which SML and HTMLDiff were installed, So it is not platform independent.

d) Not Scalable for Large Community of Users: - this version is also not scalable only one user can access the system.

We therefore choose to re-implement HtmlDiff, a more recent comparison algorithm developed by rohland [4], using asp.net, and side-by-side display of changes in the newer version. In this paper we report the design and implementation of Htmldiff.net.

2. LITERATURE REVIEW

The AT&T Internet Difference Engine (AIDE), it is a tool for tracking and viewing changes on the web [1].It provides "personalized" view of versions of www pages with three tools within it.

a) W3newer: It is a tracking tool. It regularly accesses the W3 to catch when pages on a user's bookmarks have changed.

b) Snapshot: It concedes a user to download versions of a page or flat file.

c) htmldiff: It examine two versions of HTML pages and build a "merged" page to show the changes.

AIDE allows users to supply URLs of pages of concern, whom it then checks for changes. It files versions of pages users see, in two ways either upon an explicit request or more commonly by saving each new version as it becomes available and then recording which versions a user actually sees. By default, a user is shown the differences between the current version of a page and the last version seen by the user.

A number of systems are now available to track when pages are modified [5] appears to be the most widely known example. But most of them do not tell in detail how a particular page has changed. There are only a few systems that support the comparison of web pages at the HTML level. Human-readable comparisons of HTML pages are one way to represent the differences between versions. Another way is a binary-level comparison, that will typically run faster and encode more efficiently. In TopBlend a differencing tool Vo's vdelta program [6] was used to compute a delta-encoding of two versions of a page for comparing its performance with a best-case binary-level comparison. Elliot Berk of Princeton University re-implemented HtmlDiff in Java [7], again using Hirschberg's algorithm [1], but the prototype was not fully functional. Like Hirschberg’s algorithm, the Jacobson-Vo algorithm was worst-case quadratic time, but the Jacobson-Vo algorithm has the nice property that the worst cases are very unusual and it typically will run in linear time. One is AIDE; another is WebBeholder [8], that also uses an HtmlDiff-like tool and an agent community to find and display changes on the World Wide Web; and the last is WebIntegrity [9], that displays the differences between two versions of a page on a site using frames to display the old and new versions side-by-side.

3. OBJECTIVE OF THE RESEARCH

The main objective of the web page change is to show the changes in the two different versions of the web pages or two flat files. The user can directly paste the source code of two web pages or text of two flat files or can input the URL or URI of the webpage or flat files.

- Addition of new content
- Deletion of contents
- Detecting changes at multiple locations in webpage or flat file.

4. PROPOSED WORK

The architecture of change detection in web page or flat files can detect changes in the two versions. The user can directly download the source code or contents of webpage or flat file by input the URLs. The user can also choose to input the source code or content of file directly to the textboxes. The architecture uses htmldiff algorithm for comparing the two different versions. The changes are shown by insertion or deletion in a label separately.

The whole system is designed in asp.net and programmed in c#.net. Following are the methods used for comparing two versions of a web page or flat file:

A) If the input is URL of two versions

(i) Input the URL of Old and New webpage or flat file.

(ii) Download the source code or text of file by class object of webclient().

(iii) Parse the webpage or text and make string array for both the versions as OldText[] or NewText[].

(iv) Pass the both string array as argument in HtmlDiff Algorithm.

(iv) Show the difference between two versions by HtmlDiff() on the label.

b) If the input is the source code of web page or contents of flat file.

(i) Input the source code or text of the file.

(ii) Parse the webpage or content and make string array for both the versions as OldText[] or NewText[].

(iii) Pass the both string array as argument in HtmlDiff Algorithm.

(iv) Show the difference between two versions by HtmlDiff() on the label.

Flow chart of methods of page change detection.

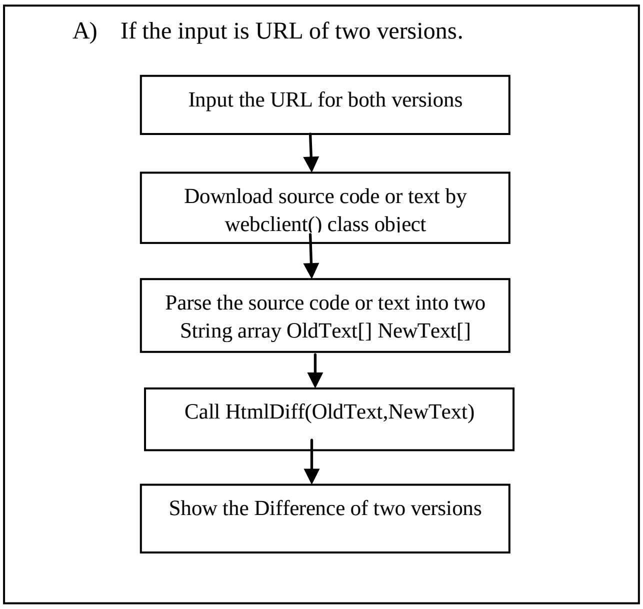

A) If the input is URL of two versions.
Input the URL for both versions
Download source code or text by webclient() class object
Parse the source code or text into two String array OldText[] NewText[]
Call HtmlDiff(OldText,NewText)
Show the Difference of two versions

Fig 3.1 Input Is URL of Two Versions

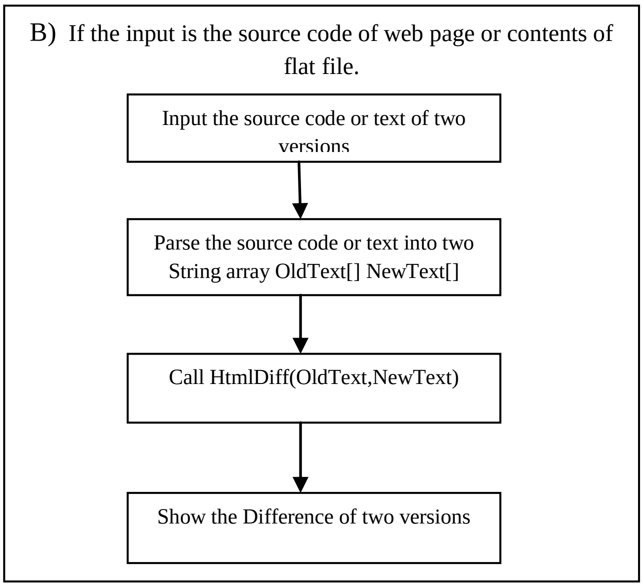

B) If the input is the source code of web page or contents of flat file.
Input the source code or text of two versions
Parse the source code or text into two String array OldText[] NewText[]
Call HtmlDiff(OldText,NewText)
Show the Difference of two versions

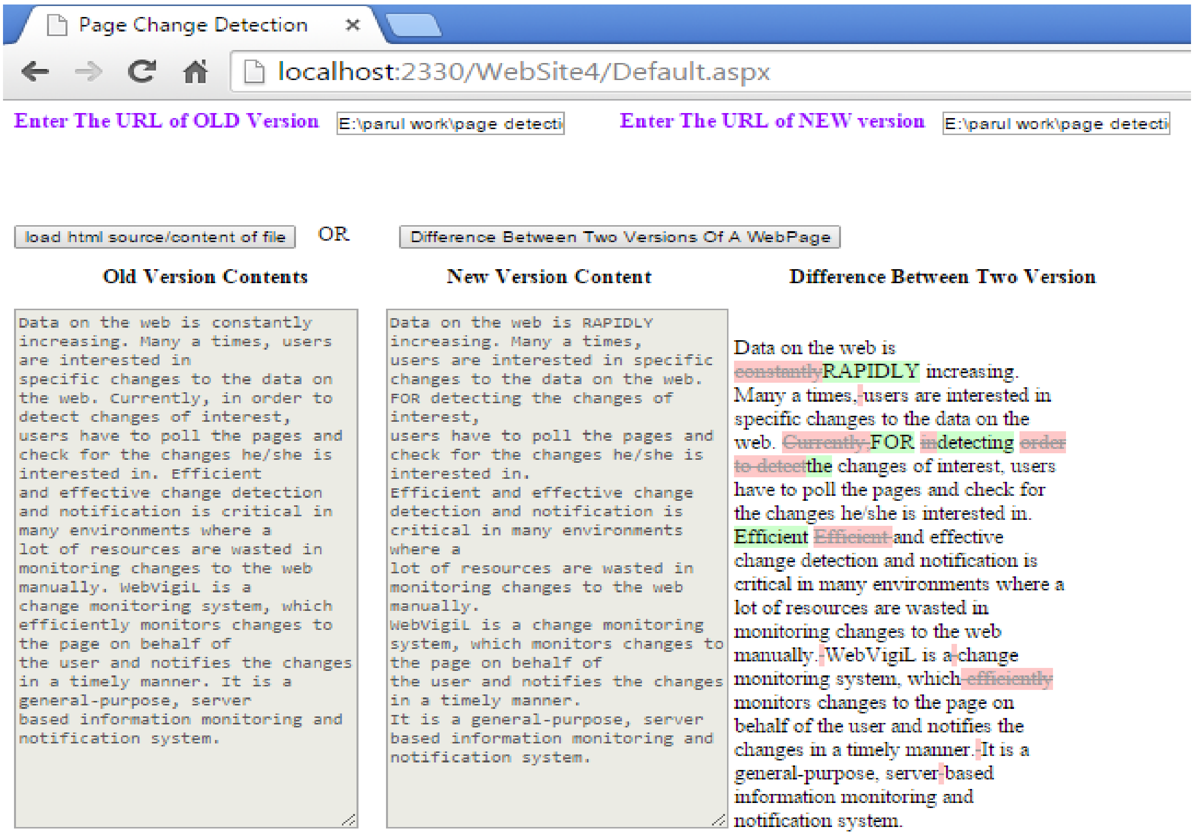
Fig 3.2 Input Is Source code of webpage or content of Two Version

5. RESULT

These are the Results for different inputs

A) Addition or Deletion of text in flat file by input URL

Here input is two URL of two version of a flat file in this result [Fig. 4.1] the changes are shown.



Page Change Detection
localhost:2330/WebSite4/Default.aspx
Enter The URL of OLD Version
E:\parul work\page detecti
Enter The URL of NEW version
E:\parul work\page detecti
load html source/content of file
OR
Difference Between Two Versions Of A WebPage

Old Version Contents

Data on the web is constantly increasing. Many a times, users are interested in specific changes to the data on the web. Currently, in order to detect changes of interest, users have to poll the pages and check for the changes he/she is interested in. Efficient and effective change detection and notification is critical in many environments where a lot of resources are wasted in monitoring changes to the web manually. WebVigiL is a change monitoring system, which efficiently monitors changes to the page on behalf of the user and notifies the changes in a timely manner. It is a general-purpose, server based information monitoring and notification system.

New Version Content

Data on the web is RAPIDLY increasing. Many a times, users are interested in specific changes to the data on the web. FOR detecting the changes of interest, users have to poll the pages and check for the changes he/she is interested in. Efficient and effective change detection and notification is critical in many environments where a lot of resources are wasted in monitoring changes to the web manually. WebVigiL is a change monitoring system, which monitors changes to the page on behalf of the user and notifies the changes in a timely manner. It is a general-purpose, server based information monitoring and notification system.

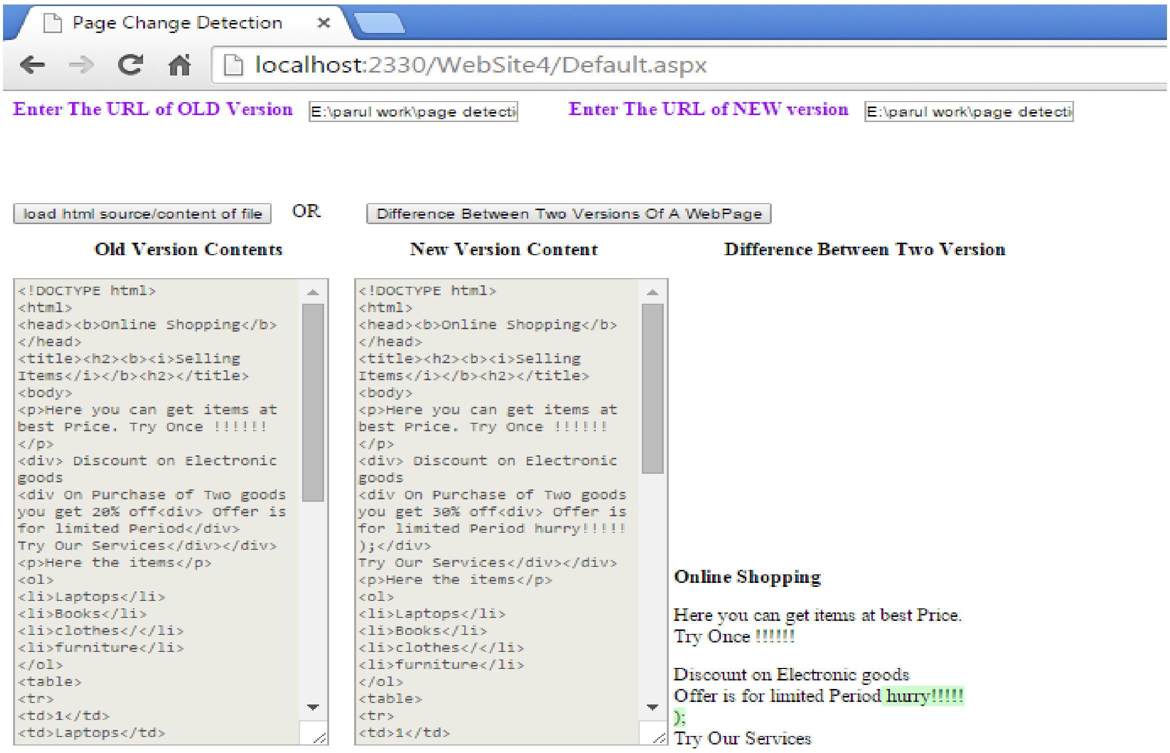
Difference Between Two Version

Data on the web is ~~constantly~~RAPIDLY increasing. Many a times, users are interested in specific changes to the data on the web. ~~Currently,~~FOR ~~in~~detecting ~~order to detect~~the changes of interest, users have to poll the pages and check for the changes he/she is interested in. Efficient ~~Efficient~~ and effective change detection and notification is critical in many environments where a lot of resources are wasted in monitoring changes to the web manually. WebVigiL is a change monitoring system, which ~~efficiently~~ monitors changes to the page on behalf of the user and notifies the changes in a timely manner. It is a general-purpose, server based information monitoring and notification system.

Fig 4.1 Addition or Deletion of text in flat file

B) Addition In Html WebPage by providing URL of both versions

Here input is two URL of two versions of a webpage in this result [Fig. 4.2] shown that what new contents are added to the old version.


Page Change Detection
localhost:2330/WebSite4/Default.aspx
load html source/content of file
OR
Difference Between Two Versions Of A WebPage

Enter The URL of OLD Version: E:\parul work\page detecti

Enter The URL of NEW version: E:\parul work\page detecti

Old Version Contents

```
<!DOCTYPE html>
<html>
<head><b>Online Shopping</b>
</head>
<title><h2><b><i>Selling
Items</i></b><h2></title>
<body>
<p>Here you can get items at
best Price. Try Once !!!!!!
</p>
<div> Discount on Electronic
goods
<div On Purchase of Two goods
you get 20% off<div> Offer is
for limited Period</div>
Try Our Services</div></div>
<p>Here the items</p>
<ol>
<li>Laptops</li>
<li>Books</li>
<li>clothes</</li>
<li>furniture</li>
</ol>
<table>
<tr>
<td>1</td>
<td>Laptops</td>
```

New Version Content

```
<!DOCTYPE html>
<html>
<head><b>Online Shopping</b>
</head>
<title><h2><b><i>Selling
Items</i></b><h2></title>
<body>
<p>Here you can get items at
best Price. Try Once !!!!!!
</p>
<div> Discount on Electronic
goods
<div On Purchase of Two goods
you get 30% off<div> Offer is
for limited Period hurry!!!!!
);</div>
Try Our Services</div></div>
<p>Here the items</p>
<ol>
<li>Laptops</li>
<li>Books</li>
<li>clothes</</li>
<li>furniture</li>
</ol>
<table>
<tr>
<td>1</td>
```

Difference Between Two Version

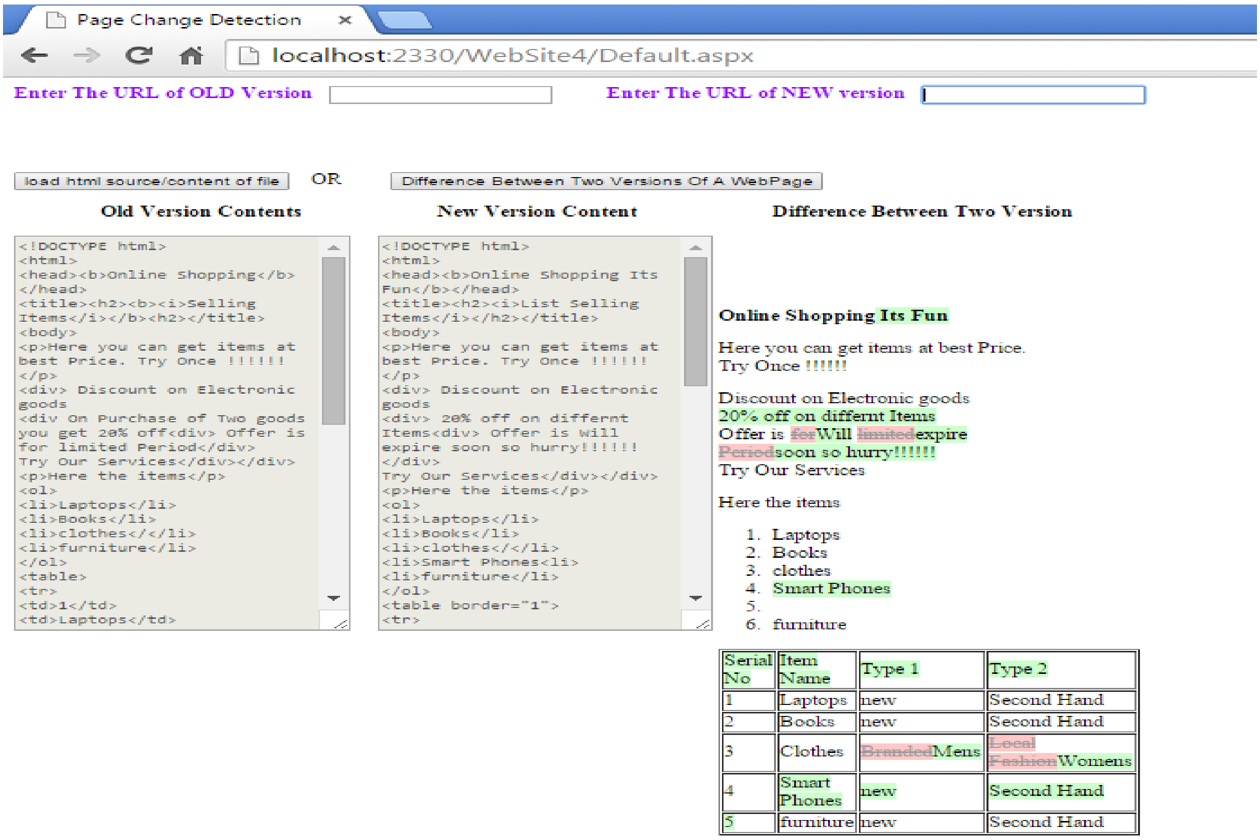
Online Shopping

Here you can get items at best Price.
Try Once !!!!!!

Discount on Electronic goods
Offer is for limited Period hurry!!!!!
);
Try Our Services

Fig 4.2 Addition of Contents In Html WebPage

c) Deletion and Addition providing by html page source code
Here input is two version's web page source code is directly input to the TextBox and difference is shown in the [Fig. 4.3]


Page Change Detection
localhost:2330/WebSite4/Default.aspx
load html source/content of file
OR
Difference Between Two Versions Of A WebPage

Enter The URL of OLD Version

Enter The URL of NEW version

Old Version Contents

```
<!DOCTYPE html>
<html>
<head><b>Online Shopping</b>
</head>
<title><h2><b><i>Selling
Items</i></b><h2></title>
<body>
<p>Here you can get items at
best Price. Try Once !!!!!!
</p>
<div> Discount on Electronic
goods
<div On Purchase of Two goods
you get 20% off<div> Offer is
for limited Period</div>
Try Our Services</div></div>
<p>Here the items</p>
<ol>
<li>Laptops</li>
<li>Books</li>
<li>clothes</</li>
<li>furniture</li>
</ol>
<table>
<tr>
<td>1</td>
<td>Laptops</td>
```

New Version Content

```
<!DOCTYPE html>
<html>
<head><b>Online Shopping Its
Fun</b></head>
<title><h2><i>List Selling
Items</i></h2></title>
<body>
<p>Here you can get items at
best Price. Try Once !!!!!!
</p>
<div> Discount on Electronic
goods
<div> 20% off on differnt
Items<div> Offer is Will
expire soon so hurry!!!!!!
</div>
Try Our Services</div></div>
<p>Here the items</p>
<ol>
<li>Laptops</li>
<li>Books</li>
<li>clothes</</li>
<li>Smart Phones<li>
<li>furniture</li>
</ol>
<table border="1">
<tr>
```

Difference Between Two Version

Online Shopping Its Fun

Here you can get items at best Price.
Try Once !!!!!!

Discount on Electronic goods
20% off on differnt Items
Offer is ~~for~~Will ~~limited~~expire
~~Period~~soon so hurry!!!!!!
Try Our Services

Here the items

1. Laptops
2. Books
3. clothes
4. Smart Phones
5.
6. furniture

Serial No	Item Name	Type 1	Type 2
1	Laptops	new	Second Hand
2	Books	new	Second Hand
3	Clothes	~~Branded~~Mens	~~Local Fashion~~Womens
4	Smart Phones	new	Second Hand
5	furniture	new	Second Hand

Fig 4.3 Deletion and Addition by html page source code

6. CONCLUSION

Due to the fast rate of change of information on www it is needed to provide change detection facilities in web pages or files. In this paper, we have proposed a technique which allows the efficient detection of web document changes.

Our approach detects changes from web page or flat files. it shows output in separate space. In Previous HtmlDiff it is shows the differences between two HTML files as a "merged" page, marking up the newer page in place with font changes to highlight the differences from the older page. But in this Html.net we show the difference separately along with the two versions webpages or flat files. For showing addition of new content green color and for showing deletion red color with strike through the text in background is used for better understanding the change in the two versions.

7. FUTURE WORK

In this paper the work for change detection between two versions of a webpage or flat file have been done. If in this, approach crawler is added. By which the files having same keyword is downloaded and indexed. Then the comparison can be performed on the new file to all downloaded file to detect the duplicity or plagiarism.

8. REFERENCES

[1] R. A. Rensink, "CHANGE DETECTION," Annu. Rev. Psychol, p. 245, 2002.

[2] F. D. H. H. a. K.-P. V. Yih-Farn Chen, "TopBlend: An Efficient Implementation of HtmlDiff in Java," Association of the Advancement of Computing in Education (AACE)., p. 1, 2000.

[3] F. e. a. Douglis, "The AT&T Internet Difference Engine: Tracking and Viewing Changes on the Web," World Wide web, 1998.

[4] T. &. D. F. Ball, "Tracking and viewing changes on the Web," in USENIX Technical Conf., 1996.

[5] D. Hirschberg, "Algorithms for the longest common subsequence problem,," Journal of the ACM, vol. 24(4), 1977.

[6] Rohland, "Rohland/htmldiff.net," https://github.com/Rohland/htmldiff.net, 7 nov 2013.

[7] Url-minder, "http://www.netmind.com/URL-minder/URL-minder.html.," 1996.

[8] D. &. V. K.-P. e. Korn, "Vdelta: Differencing and Compression, in Practical Reusable UNIX Software," John Wiley & Sons, New York, 1995.

[9] E. Berk, "HtmlDiff: A Differencing Tool for HTML Documents, Student Project," Princeton University, 1996.

[10] S. &. I. M. Saeyor, "WebBeholder: A Revolution in Tracking and Viewing Changes on The Web by agent Community," proc.WebNet98, 1998.

[11] M. K. S. Inc., "Web Integrity," http://www.mks.com/products/wi/, 1997.