# Trend of Supervised Web Data Extraction

Galih Hendro Martono
Department of Informatics
STMIK Bumigora Mataram
Doctoral Program Department of Computer Science and Electronics
Universitas Gadjah Mada

Azhari Azhari
Department of Computer Science and Electronics
Universitas Gadjah Mada

Khabib Mustafa
Department of Computer Science and Electronics
Universitas Gadjah Mada

## ABSTRACT

Website has evolved since it was first developed in 1990. Since then, the website grows rapidly. Based on the information provided by http://www.worldwidewebsize.com the number of websites is currently at least 4.54 billion pages. With a very large number, the website stores a lot of information that can be used. That problem brings up the concept of data extraction. Web data extraction aims to retrieve the contents of the website so that it can be easy to use for other purposes. The utilization of web data extraction can be used in a product catalog, news, bookstore, travel, etc. There are many systems build by different technique such as manual, supervised, un-supervised, and semi-supervised. This paper discuss supervised learning technique for web data extraction. Several previous surveys have overviewed the wrapper induction system using the concept of supervised techniques to extracted web data up to 2007. The aim of this paper is to present a comprehensive overview of the research in supervised web extraction data by providing the latest research results

## General Terms

Web Data Extraction

## Keywords

Extraction web data, supervised learning, wrapper, machine learning, wrapper induction system

## 1. INTRODUCTION

Web page contains important information that is very useful such as product catalog, news, bookstore, travel, etc. Usually, web pages are written in HTML page format that can be accessed through the internet via HTTP protocol. As reported on page http://www.worldwidewebsize.com/ exposed that the current number of web pages as much as at least 4.54 billion pages. With this amount, it can be said that the website is a storehouse of information that can be utilized in various fields such as e-commerce, e-business, e-government, education, health, military, etc. To retrieve information on the pages of a website is not an easy matter considering the number of web pages that exist, so the concept of information extraction emerged. Extraction of information on the web page is to take the information contained on the website to be presented in other formats such as tables, XML, Excel, and others. Initially, the extraction of data on the web page is done by creating a program called wrapper to extract the data from the web page manually. The programs find the pattern of the web page and define rules of extraction in the wrapper. The weakness of this program is not all users can create data extraction program, the complexity of the web page considering the number of data/pages that exist on the web page and the nature of dynamic web page and template. Further development of data extraction with an automatic wrapper is done by matching and wrapping words in HTML Tags by using DOM (Document Object Model) Tree. Each tag in HTML is segmented based on HTML Tag structure such as Paragraph (P), table, List (UL), Heading (H1-H6), and so on.

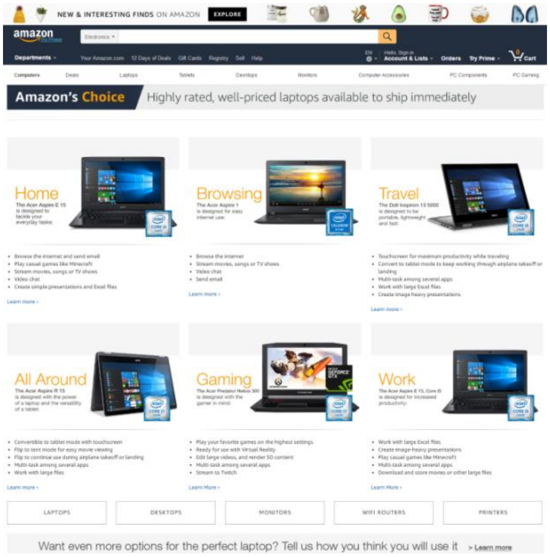
According to Zhai and Liu [1], web data extraction techniques are classified into programming wrappers, induction wrappers, and automation extractions. Meanwhile, according to Chang et.al [2] web data extraction techniques on the web page can be done manually, supervised, un-supervised, and semi-supervised. In both studies there are several terms that have the same definition, such as wrapping programming and manuals that perform data extraction by making a tool with a particular programming language, this technique can only be done by someone who has a technical background so that this technique is difficult to use by many people because it will take a lot of time and is not efficient. The systems as used in this technique are TSIMMIS [3], Minerva [4], Web-OQL [5], WW4F [6], XWRAP [7], WICCAP [8], and Wargo [9]. Wrapper Induction and supervision is an extraction technique using machine learning techniques using labeled data and objective function [10]. This technique takes a longer time for data extraction due to labeling process, training and the user has to decide whether the target is either a page or non-tag text on a web page. Such system used this technique are WIEN [11], Softmealy [12], Stalker [13], WL [14], IDE [15], SRV [16], RAPIER [17], WHISK [18], etc. Research conducted by [19]–[21] using a learning machine approach resulted in wrapping. Automation extraction is done unattended on its own by using an existing wrapping system and no labeling training is required. Users can only enter the URL of the web page and the system will extract the web page. Systems using this technique are RoadRunner [22], EXALG [23], DeLa , Trinity [24] and DEPTA [25]. Chang et al [2] formulated another technique for the extraction of semi-supervised web data. This technique performs data extraction based on the extraction of generating rules. However, this technique does not need to be labeled training data. The systems used in this technique are IEPAD [26], OLERA [27], and Thresher [28]. Chang et al [2], [29] noted that the basic content of data extraction techniques on the web is based on the wrapping phase: collecting training pages, labeling, generalization extraction rules, extracting relevant data, and outputs in any format (eg DBMS or XML format).

This paper aim is to provide a comprehensive overview of the development and trend in a supervised web data extraction systems and the differences from one technique to another. This paper will discuss the previous survey of supervised techniques on data extraction conducted up to 2007, then will be discussed the development of other systems related to the same thing starting from 2008 to see the development and trend of supervised learning techniques in doing data extraction on the web. To evaluate the data extraction system used three dimensions proposed by Chang [29] i.e task difficulties, atumation degree, and techniques used.

# 2. THEORY

## 2.1 Web Data Extraction

Web data extraction is the process of extracting data from web page to get valuable information about the content of web page and translate it into structured format. Web data extraction needed because not all the information on the web page has useful information, such as ads, videos, comment, banners and more. The results of data extraction can be presented in other formats such as tables that describe the content of the web page so that it is more convenient to see or can be used in data integration. Data on the extraction of web pages can be used as inputs within the system or be using data extraction as a database in other systems. An example is an e-commerce web page that can be seen in Figure 1.



Amazon's Choice Highly rated, well-priced laptops available to ship immediately

Home
The Acer Aspire E 15 is designed to tackle your everyday tasks:

- Browse the internet and send email
- Play casual games like Minecraft
- Stream movies, songs or TV shows
- Video chat
- Create simple presentations and Excel files

Learn more ›

Browsing
The Acer Aspire 1 is designed for easy internet use:

- Browse the internet
- Stream movies, songs or TV shows
- Video chat
- Send email

Learn more ›

Travel
The Dell Inspiron 13 5000 is designed to be portable, lightweight and fast:

- Touchscreen for maximum productivity while traveling
- Convert to tablet mode to keep working through airplane takeoff or landing
- Multi-task among several apps
- Work with large Excel files
- Create image heavy presentations

Learn more ›

All Around
The Acer Aspire R 15 is designed with the power of a laptop and the versatility of a tablet:

- Convertible to tablet mode with touchscreen
- Flip to tent mode for easy movie viewing
- Flip to continue use during airplane takeoff or landing
- Multi-task among several apps
- Work with large files

Learn more ›

Gaming
The Acer Predator Helios 300 is designed with the gamer in mind:

- Play your favorite games on the highest settings
- Ready for use with Virtual Reality
- Edit large videos, and render 3D content
- Multi-task among several apps
- Stream to Twitch

Learn More ›

Work
The Acer Aspire E 15, Core i5 is designed for increased productivity:

- Work with large Excel files
- Create image-heavy presentations
- Play casual games like Minecraft
- Multi-task among several apps
- Download and store movies or other large files

Learn More ›

Want even more options for the perfect laptop? Tell us how you think you will use it › Learn more

**Fig. 1 Amazon.com**

The resulting data extraction from web pages can be seen in Table 1.

**Table 1 Extracted Web**

| *Merk* | *Type* | *Processor* | *RAM* | *Hard disk* | *Graphics* | *OS* | *Colour* | *Customer Reviews* | *Price* |
|---|---|---|---|---|---|---|---|---|---|
| *Acer Aspire* | *E 15 E5-575-33BM* | *Intel Core i3-7100U 7th Generation* | *4GB DDR4* | *1TB 5400 RPM* | *Intel HD Graphics 620 15.6-Inch* | *Windows 10 Home* | *Obsidian Black* | *3.6* | *$349.99* |
| *Acer Aspire* | *A114-31-C4HH* | *Intel Celeron N3450* | *4GB* | *32GB* | *Intel HD Graphic 1.14Inch* | *Windows 10 Home* | *-* | *3.7* | *$199.99* |
| *Dell Inspiron* | *i5379-5043GRY-PUS* | *Core i5-8250U* | *8GB* | *1TB* | *13.3Inch Touch Display* | *-* | *-* | *3.4* | *$649.99* |
| *Acer Aspire* | *R5-571TG-7229* | *Intel Core i7 7th Generation* | *12GB DDR4* | *256GB SSD* | *GeForce 940MX 15.6Inch Full HD Touch* | *-* | *-* | *4.0* | *$699.99* |
| *Acer Predator* | *G3-571-77QK* | *Intel Core i7 7700HQ* | *16GB DDR4* | *256GB SSD* | *GeForce GTX 1060-6GB* | *-* | *Red Backlit KB, Metal Chasis* | *3.9* | *$999.99* |
| *Acer Aspire* | *E5-576G-5762* | *Intel Core i5-8250U* | *8GB* | *256GB SSD* | *GeForce MX 150* | *-* | *-* | *4.1* | *$599.00* |

In the table 1 can be seen that by doing data extraction on the web page, we can easily see the contents of the web page as a whole. An example of a web data extraction study for the case of data integration in e-commerce has been done by Silva and Cardoso [30] to resolve the data source heterogeneity problems and offer the advantages of using common shared structural format represented with an ontology. Data extraction research on e-commerce can be used as a further analysis such as analyzing the activity of its competitors [31]. Several other studies have performed web data extraction such as [32]–[40].

## 2.2 Document Object Model (DOM) Tree

DOM Tree is language independent application programming interface for accessing and manipulating HTML and XML documents as a tree structure. With DOM Tree, the programmer can easy to build document, navigate the web structure, and modification elements and content in the tree-like structure [41]. DOM Tree represented every HTML tag as an object. Figure 2 shows an example of an HTML code.

```
<!DOCTYPE HTML>
<html>
<head>
  <title>amazon.com</title>
</head>
<body>
  Laptops
</body>
</html>
```

**Fig. 2 Example of HTML Code**

At web document structure, tag <html> is the root, then <head> and <body> are its children. Every tag in HTML as a nodes element or elements. Generally, the structure in HTML has the same structure. Each node in the DOM Tree displayed information about the corresponding HTML element. Few tag in the website can be seen in Table 2.

**Table 2 HTML Element**

| Tag | Description |
|---|---|
| <!DOCTYPE> | Defines the information document type |
| <html> | Defines an HTML document |
| <head> | Defines information about the document such as page’s title, description, and verification scrips |
| <title> | Defines a title of document |
| <body> | Defines the document body |
| <h1> to <h6> | Defines HTML headings |
| <p> | Defines a paragraph |
| <br> | Insert a single line break |
| <img> | Defines an image |
| <a> | Defines a hyperlink |
| <table> | Defines a table |
| <th> | Defines a header cell in a table |
| <tr> | Defines a row in a table |
| <td> | Defines a cell in a table |

Every HTML tag in web pages described in form of DOM Tree as seen in Figure 3.

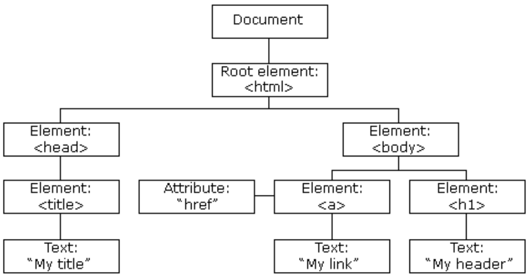


**Fig. 3 DOM Tree**

The use of DOM Tree in data extraction is done by looking at the structure and relationship of existing nodes to define relevant data [32]. In 2003, Liu et al [42] introduced the concept of generalized nodes to group the same data in a similar region in the DOM Tree web page. This is done based on the assumption that existing data is presented in the same format and repeated. Zhai [25] developed the concept in [42] that the structure in the DOM Tree not only presents recurring data but also has the same structure and looks for a common strata tree by using a tree matching algorithm. The weakness of DOM Tree approach to extraction web data is not generally applicable for data extraction. It fails to disjunctive and optional data [32]. To solve this problem, another approach used to perform data extraction is used the visual properties of data, such as visual boundary, text color, and size. Examples of studies that use visual cue can be viewed in [43].

## 2.3 Supervised Extraction Web Data

Supervised web data extraction is a method of extraction of web data by using labeling data and training data. Data extraction technique is done manually that is by providing labeling of data and make the program to do extract data on the web. Figure 4 represents the process of supervised techniques on web data extraction[2], [29].

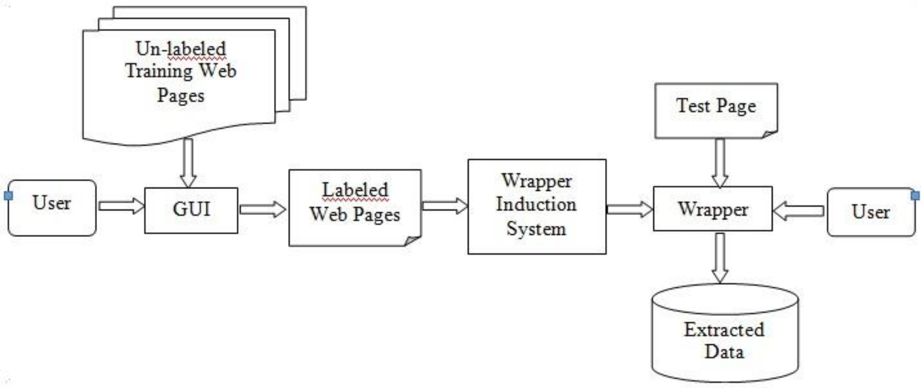


**Fig. 4 Supervised Wrapper Induction System**

Figure 4 describes data extraction with supervised techniques performed when a user with GUI labeling data from previously unlabeled data to extract data with the wrapper induction system so that the website is extracted. Wrapper is the process of extracting data from unstructured website pages (HTML) into structured data (XML). The extraction process with supervised learning works by extracting data by using unlabeled data. On the website, there are millions of data in the form of unlabeled data so it needs a technique to convert unlabeled data into labeled data [44].

# 3. RELATED WORKS

There are several papers that have conducted surveys related to the implementation of supervised techniques to perform web data extraction as in [2], [29], [45], [46]. The paper describes several systems that implement supervised techniques such as WIEN [11], Softmealy [12], Stalker [13], WL [14], IDE [15], SRV [16], RAPIER [17], WHISK [18]. Kushmerick [11] developed various types of wrappers based on inductive learning called WEIN. This tool takes a set of pages as input and learning from the set of pages. WEIN studies every word on the website and generalizes the rule to take a conclusion. In the implementation built a class HLRT (head-left-right-tail) to perform scanning on each body of page and separate between head and tail. Similar to WIEN, Softmealy [12] generates a rule but uses the automata concept called finite-state transducer (FST) and contextual rules. FST consists of two parts: body transducer and tuple transducer. Body transducer extracts the sub-strings of the page that have the tuples. The tuple transducer performs the tuple extraction from the body. Contextual rules are used to recognize the context of separators between adjacent attributes so it can deal with diverse structuring patterns and maintain parsing efficiency. The built system can be used for semi-structured web extraction containing missing attributes, attributes with multiple values, variant attribute permutations, exceptions, and typos.

Stalker working based on inductive learning algorithm. The idea is to build an embedded catalog (EC) formalism in the form of a tree where the leaves extract the attributes and the internal nodes are list of tuples. WL does a lesson called "builder". Each "builder" is a restricted L language by generating L1 and L2. Unlike WIE, Softmealy, and Stalker using inductive learning to do training examples, the IDE used an instance-based method by comparing each new instance (page) with labeled instances (pages) has been saved previously so it does not require labeled page to learn extraction rules. Labeling need when the page cannot be extracted. The users give labels or mark the target items in a set of training pages and the system learns from extract target items from another page. Extraction rules have two parts: a prefix marking the beginning of the target item and the suffix for marking the ending of a target item. The process of extraction of web data on IDE is done by calculating similarity measure between target items that have been labeled with a new page, this is done by calculating similarity measure on prefix and suffix. Another development of web data extraction with supervised techniques using inductive logic programming as in SRV and RAPIER.

Inductive logic programming is an intersection of inductive learning and logic programming [47]. The concept of inductive learning is used to construct hypotheses from observations (examples) and build new knowledge from experience while logic programming used to representational mechanism of hypothesis. Soderland [18] proposed WHISK to handle extraction rules from structured text fo free text. The user selected of instance to be tagged from a given set of training example then the system uses the tagged instance to create rules and test the accuracy of the proposed rules. The rules are based on regular expression that has two component; identify the context that makes a phrase relevant and determines the exact delimiters of the phrase to be exact. WHISK can extract several records from a document.

Based on previous research it is known that the learning process is done by inductive learning, instance based method, and inductive logic programming. For the type of data can be structured text, semi structured text, un-structured text, and free text. In each system there are disadvantages advantages, for example, support for data extraction with multiple-slot that can be used for data extraction from multi-source (multiple sites). Table 3 summarizes the web data extraction system with supervised learning techniques.

**Table 3 Comparison Wrapper Induction System**

| System Name | Extraction Rule Learning | | | | Text Style | | | | Multi-slot |
|---|---|---|---|---|---|---|---|---|---|
| | **Inductive Learning** | **Instance Based** | **Inductive Logic Programming** | **Other** | **Structured Text** | **Semi structured Text** | **Unstructured Text** | **Free Text** | |
| WIEN | √ | | | | √ | √ | | | No |
| Softmealy | √ | | | | | | √ | | No |
| Stalker | √ | | | | | √ | | | No |
| WL | | | | √ | √ | | | | Yes |
| IDE | | √ | | | √ | | | | No |
| SRV | | | √ | | | √ | | | No |
| RAPIER | | | √ | | | √ | | | No |
| WHISK | | | | | √ | √ | | √ | Yes |

# 4. DISCUSSION

Web data extraction using supervised techniques has been studied by many researchers. Based on [2], [29], [45], [46] which conducted several surveys in the extraction of web data with supervised techniques it is known that the development of wrapper induction system was done in 1997-2007. Initially, the development of web data extraction with supervised technique is done by inductive learning method, instance based, and inductive logic programming as in table 3. The next development, extraction systems used machine learning method in the learning process on data labeling. Studies related with the implementation of supervised learning techniques on web data extraction only found four studies since 2008 that are [48]–[51].

This section will review comparison implementation supervised learning techniques for extracting data and discuss evaluate extraction system based on [29].

## 4.1 Comparison Supervised Learning Techniques for Extraction Data

Some researchers use the SVM method to classify text on web pages such as Yusuf, Othman, and Salim [48] devised a system to classify the web documents using the Support Vector Machine (SVM) and compare them with four types of kernels such as Radial Basis Function (RBF), linear, polynomial, and sigmoid. Linear kernel techniques show the best in web document classification. The researchers have built Web Resources Classification System (WRCS) to extract the web information such as metadata, contents, links, and image. To validate the classification accuracy and effectiveness of the applied algorithm, it’s used the global data set and the local data set. The Global data set collected from the Bank Search Information Consultancy Ltd and University of Reading and the Local data set used from Bursa Malaysia. The results show that the best kernel usage is the linear kernel with accuracy percentage of 89.80% for the Global Data Set and 75.36% for The Local Data Set. Ahmad et al [49] presented SVM classification and machine learning techniques to satire detection. For obtaining the results, the researchers used TF-IDF-BNS feature selection. The data used consisted of 2624 newswire and 171 satire news articles. The SVM method used is combined with several weighing parameters such as Frequency Frequency-Inverse Document Frequently (TF-IDF), Term Frequency Bi-normal separation feature scaling (TF-BNS), Binary Feature Weight (BIN), TF-BNS-IDF the combination of SVM with TF-BNS-IDF has high precision (82.6%), high recall (87%), and high F-Score (84%) when compared to others. Ali and Omar [50] proposed several machine learning approaches such as linear logistic regression, linear discriminant analysis, and support vector machine to extraction Arabic key phrase. The results showed that SVM achieved the best performance (precision 86.67%) compared to other methods. Different from other researchers, Mayilvaganan and Sakthivel [51] build system using Bayesian Classification, Expected Maximization algorithm and IF-THEN rules method of rule induction. Researchers raised the case of online book sales in the study. IF-THEN rules method is used to facilitate users to find information on website i.e "book title" or "author name". The search is done based on the rule established by the IF-THEN rule, when the information is not available on this website then the extraction will be continued by using Bayesian Classification. While expected maximization is used to make sure that the extracted information is are correct. The research also implements the Internet Intelligent Agent System to solving the problem of integrating data from multiple source websites so that the user in finding information about "book title" or "author name" uses only an interface. Table 4 below shows the comparison and feature of different supervised learning techniques.

**Table 4 Comparison and Feature of Different Supervised Learning Techniques**

| Authors | Description | Method | Benefit | Result |
|---|---|---|---|---|
| Yusuf, Othman, and Salim (2010) | Extracted web document and compare with Radial Basis Function (RBF), linear, polynomial, and sigmoid | SVM | Web Resources Classification System (WRCS) can extract the web information such as metadata, contents, links, and image | The linear is the best kernel with accuracy percentage of 89.80% for the Global Data Set and 75.36% for The Local Data Set |
| Mayilvaganan and Sakthivel (2013) | Extracted web data from multiple sites | Bayesian Classification, Expected Maximization algorithm, and IF-THEN rules method | Users can fast access multiple sites from one interface using intelligent agent system and help for searching data in multiple sites | - |
| Ahmad et al (2014) | Extracted news online for satire detection using 2624 newswire and 171 satire news article | SVM and TF-IDF-BNS | Determining whether a newswire article is satire or no | Precision 82.6%, Recall 87%, and F-Score 84% |
| Ali and Omar (2014) | Compared several machine learning such as linear logistic regression, linear discriminant analysis, and support vector machine to extraction Arabic key phrase | SVM | Building automatic keyphrase generating system | Precision 86.67% Recall 90.01% F-measure 88.31% |

## 4.2 Evaluating Extraction System

To evaluate the extraction systems at [48]–[51], in this paper used the criteria proposed by [29] there are based on task difficulties, automation degree, and techniques used. Task difficulties classified into page type (PT), Non-HTML support (NHS), Singeleton Pages (SP), Extraction Level (EL), Nested Data-Object (Nested), Missing Attributes (MA), Multi-Valued Attributes (MVA), Multi-Ordering Attributes (MOA), Fixed/Variant-Format Attributes (FVF), UnTokenized Attributes (UTA), Sequential-Pattern Attributes (SPA), UnDistingushable Attribues (UDA). Automation degree based classified into GUI Support, Crawler Support, Output Support, Training Examples, and API Support. The Technique used based classified into Scan Pass, Extraction Rule Type, Features Used (FU), Learning Algorithm, and Tokenization Schemes. In this paper not all a set of criteria are used because of the limited information available in the papers reviewed. In the task difficulties dimension, criteria used are page type (PT), Non-HTML support (NHS), Singeleton Pages (SP), Extraction Level (EL), and Sequential-Pattern Attributes (SPA). In the automation degree dimension, criteria used are GUI Support and training example. While in the techniques used dimension, criteria used are Features Used (FU), learning algorithm, and tokenization schemes.

### *4.2.1 Task Difficulties Dimension*

**Page Type (PT):** Some extraction systems using file formats such as structured text, semi-structured text, and free text as inputs in the data extraction process.

**Non-HTML Support (NHS):** This criterion describes the system support HTML document or not.

**Singleton Pages (SP):** On the static website, the layout of each page will be the same so it easier to extracted data from the site. But, in some cases, it is possible the layout of the website different on each page, so that the extraction systems need to recognize layouts and templates on the website.

**Extraction Level (EL):** Extraction of data on the website can be done at various levels including field level, record level, page level, and site level.

**Sequential-Pattern Attribute (SPA):** Some extraction system can handle the delimiter of an attribute as a single structure and other can handle delimiter as a set of sequential delimiters.

### *4.2.2 Automation Degree Dimension*

**GUI Support:** Help user manage the process of extraction data by providing user-friendly interface in the example the labeling process, the input process, and selected target.

**Training Example:** The supervised learning need training example to labeled data as shown in fig. 4.

### *4.2.3 Techniques Used Dimension*

**Features Used (FU):** Describe the process extraction data from website. Mostly, some systems used HTML Tags and DOM Tree for features in extraction website and other, SRV [16] used orthographic features, token's length, and link grammars.

**Learning Algorithm:** Earlier extraction system used inductive learning, instance based method, and inductive logic programing to generate extraction rules for training example. Recently, some of machine learning method used to generate extraction rules.

**Tokenization Schemes:** To cleaning data from punctuation, tag HTML, delimiter, etc. Tokenization is done by dividing the sentence, paragraph on the document into small part (tokens)

**Table 5 Evaluation of Wrapper Induction Systems Based on Supervised Learning**

| **Authors** | **Task Difficulties** | | | | | **Automation Degree** | | **The Techniques Used** | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **PT** | **NHS** | **SP** | **EL** | **SPA** | **GUI Support** | **Training Example** | **FU** | **Learning Algorithm** | **Tokenization Schemes** |
| Yusuf, Othman, and Salim (2010) | Structured Text | Yes | Yes | Site Level | Yes | Yes | Labeled | HTML Tags | SVM | Tag Level |
| Mayilvaganan and Sakthivel (2013) | Structured Text | Yes | No | Site Level | Yes | Yes | Labeled | HTML Tags/Dom Tree | Naive Bayesian | Tag Level |
| Ahmad et al (2014) | Structured Text | Yes | Yes | Site Level | Yes | Yes | Labeled | HTML Tags | SVM | Word Level |
| Ali and Omar (2014) | Structured Text | Yes | Yes | Record Level | Yes | Yes | Labeled | HTML Tags | SVM | Tag Level |

## 5. CONCLUSIONS

**Conclusions.** This paper provides a brief summary of implementation supervised learning techniques for extraction web data since 2008. From the results of this survey, it is known that research related to supervised extraction of web data is still little to be found, this is understandable because this method is very expensive and time-consuming from the user to do the labeled web page. Supervised learning needs a large of labeled data for training data because the amount of training data (website) greatly affects the amount of vocabulary is produced.

Supervised learning technique is still related to unlabeled training data by using machine learning algorithm. SVM method is widely used to perform unlabeled training data with outcome precision above 75%. SVM method is a robust algorithm that could facilitate high dimensions. Furthermore, SVM method has the highest predictive accuracy compared to another method [52]–[54]. To get a better accuracy in the classification process, it can be combined with other kernel function such as polynomial, Gaussian Radial Basis Function, Exponential Radial Basis Function, Multi-Layer Perceptron, Fourier Series, Splines, B Splines, Additive Kernels, Tensor Product, Hyperbolic Tangent (sigmoid), Fourier Series, etc. Kernel functions are very important in improving the performance of classifiers [55].

**Future Works.** Next studies of data extraction with supervised techniques can use other machine learning methods such as Naïve Bayes, K-Nearest Neighbors, decision tree or hybrid algorithm. In addition, web data extraction can be done based on semantic web and XML.

## 6. REFERENCES

[1] Y. Zhai and B. Liu, "Structured Data Extraction from the Web Based on Partial Tree Alignment," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 12, pp. 1614–1628, 2006.

[2] Chia-Hui Chang, M. Kayed, M. R. Girgis, and K. F. Shaalan, "A Survey of Web Information Extraction Systems," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 10, pp. 1411–1428, 2006.

[3] J. Hammer, J. McHugh, and H. Garcia-Molina, "Semistructured data: the TSIMMIS experience," in *In Proceedings of the 1st East-European Symposium on Advances in Databases and Information Systems (ADBIS)*, 1997, pp. 1–8.

[4] V. Crescenzi and G. Mecca, "Grammars have exceptions," *Inf. Syst.*, vol. 23, no. 8, pp. 539–565, 1998.

[5] G. O. Arocena and A. O. Mendelzon, "WebOQL: Restructuring documents, databases, and webs," in *Proceedings of the 14th IEEE International Conference on Data Engineering (ICDE), Orlando, Florida*, 1998, vol. 5, no. 3, pp. 24–33.

[6] A. Sahuguet and A. Fabien, "Building Intelligent Web Applications Using Lightweight Wrappers," *Data Knowl. Eng.*, pp. 283–316, 2001.

[7] L. Liu, C. Pu, and W. Han, "XWRAP: an XML-enabled wrapper construction system for Web information sources," *Proc. 16th Int. Conf. Data Eng. (Cat. No.00CB37073)*, no. February, pp. 611–621, 2000.

[8] Z. Li and W. N.k, "WICCAP : From Semi-structured Data to Structured Data," in *Proc. 14th Int'l Conf. World Wide Web*, 2004, pp. 66–75.

[9] J. Raposo, A. Pan, M. Alvarez, J. Hidalgo, and A. Vina, "The Wargo system: Semi-automatic wrapper generation in presence of complex data access modes," *Proc. - Int. Work. Database Expert Syst. Appl. DEXA*, pp. 313–317, 2002.

[10] Y. Kim and S. Lee, "SVM-based web content mining with leaf classification unit from DOM-tree," in *2017 9th International Conference on Knowledge and Smart Technology: Crunching Information of Everything, KST 2017*, 2017, pp. 359–364.

[11] N. Kushmerick, "Wrapper Induction for Information Extraction," in *Proceedings of the Fifteenth International Conference on Artificial Intelligence (IJCAI)*, 1997, pp. 729–735.

[12] C. N. Hsu and M. T. Dung, "Generating finite-state transducers for semi-structured data extraction from the Web," *Inf. Syst.*, vol. 23, no. 8, pp. 521–538, 1998.

[13] I. Muslea, S. Minton, and C. Knoblock, "A Hierarchical Approach to Wrapper Induction," in *Proceedings of the Third International Conference on Autonomous Agents*, 1999.

[14] W. W. Cohen, M. Hurst, W. W. Cohen, M. Hurst, L. S. Jensen, and L. S. Jensen, "A flexible learning system for wrapping tables and lists in HTML documents," *Proc. 11th Int. Conf. World Wide Web*, no. July, pp. 232–241, 2002.

[15] Y. Zhai and B. Liu, "Extracting Web data using instance-based learning," *J. World Wide Web*, vol. 10, no. 2, pp. 113–132, 2007.

[16] D. Freitag, "Information Extraction From HTML: Application of A General Learning Approach," in *Proceedings of the Fifteenth Conference on Artificial Intelligence (AAAI-98)*, no. 0.

[17] M. E. Califf and R. J. Mooney, "Relational learning of pattern-match rules for information extraction," *Comput. Nat. Lang. Learn.*, vol. 4, pp. 9–15, 1997.

[18] S. Soderland, "Leaning Information Extraction Rules for Semi-Structured and Free Text," *J. Mach. Learn.*, pp. 233–272, 1999.

[19] N. Ashish and C. Knoblock, "Wrapper Generation for Semi-Structured Internet Sources," *SIGMOD Rec.*, pp. 8–15, 1997.

[20] B. Doorenbos, "A Scalable Comparison-Shopping Agent for the World-Wide-Web," in *In Proceedings of the First International Conference on Autonomous Agents*, 1997, pp. 39–48.

[21] C. Knoblock, K. Lerman, S. Minton, and I. Muslea, "Accurately and Reliably Extracting Data from the Web: A Machine Learning Approach," *Bull. Tech. Comm. Data Eng.*, vol. 23, no. 4, pp. 35–43, 2000.

[22] V. Crescenzi, G. Mecca, and P. Merialdo, "Roadrunner: Towards automatic data extraction from large web sites," *Proc. 27th Int. Conf. Very Large Data Bases*, pp. 109–118, 2001.

[23] A. Arasu and Garcia-Molina, "Extracting structured data from Web pages," *2003 ACM SIGMOD Int. Conf. Manag. Data*, pp. 337–348, 2003.

[24] H. A. Sleiman and R. Corchuelo, "Trinity: On using trinary trees for unsupervised web data extraction," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 6, pp. 1544–1556, 2014.

[25] Y. Zhai and B. Liu, "Web data extraction based on partial tree alignment," *Proc. 14th Int. Conf. World Wide Web - WWW '05*, pp. 76–85, 2005.

[26] C. Chang and S.-C. Lui, "IEPAD: Information Extraction Based on Pattern Discovery," *Proc. 10th Int. Conf. World Wide Web - WWW*, pp. 681–688, 2001.

[27] C. H. Chang and S. C. Kuo, "OLERA: Semisupervised Web-data extraction with visual support," *IEEE Intell. Syst.*, vol. 19, no. 6, pp. 56–64, 2004.

[28] A. Hogue and D. Karger, "Thresher: automating the unwrapping of semantic content from the World Wide Web," *Proc. 14th Int. Conf. World Wide Web - WWW '05*, no. January 2005, pp. 86–95, 2005.

[29] C. Chang, M. Kayed, M. Girgis, and K. Shaalan, "Criteria for Evaluating Information Extraction Systems," *3rd Int. Conf. Informatics Syst.*, 2005.

[30] B. Silva and J. Cardoso, "Semantic data extraction for B2B integration," in *Proceedings - International Conference on Distributed Computing Systems*, 2006.

[31] H. Chen, M. Chau, D. D. Zeng, H. Chen, M. Chau, and D. Zeng, "CI Spider : A tool for competitive intelligence on the Web CI Spider : a tool for competitive intelligence on the Web," *Deci Support Syst.*, no. April 2014, pp. 1–17, 2002.

[32] J. L. Hong, "Automated data extraction with multiple ontologies," *Int. J. Grid Distrib. Comput.*, vol. 9, no. 6, pp. 381–392, 2016.

[33] Y. Wang and Zhou L, "A Hybrid Method for Web Data Extraction.pdf," in *Web Intelligence, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on*, 2003.

[34] J. Robinson, "Data extraction from Web data sources," *Proceedings. 15th Int. Work. Database Expert Syst. Appl. 2004.*, pp. 282–288, 2004.

[35] L. P. B. Vuong, X. Gao, and M. Zhang, "Data extraction from semi-structured Web pages by clustering," in *Proceedings - 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings), WI'06*, 2006, pp. 374–377.

[36] S. Tan, J. Fan, and Y. Jiang, "Web Data Extraction Based on Label Library," in *2009 World Congress on Computer Science and Information Engineering*, 2008.

[37] H. Hong, X. Chen, G. Wu, and J. Li, "Web Data Extraction Based on Tree Structure Analysis and Template Generation," in *E-Product E-Service and E-*

*Entertainment (ICEEE), 2010 International Conference on*, 2010.

[38] N. K. Tran, K. C. Pham, and Q. T. Ha, "XPath-wrapper induction for data extraction," in *Proceedings - 2010 International Conference on Asian Language Processing, IALP 2010*, 2010, pp. 150–153.

[39] K. A. Pakojwar, R. S. Mangrulkar, and V. G. Bhujade, "Web data extraction and alignment using tag and value similarity," in *ICIIECS 2015 - 2015 IEEE International Conference on Innovations in Information, Embedded and Communication Systems*, 2015, pp. 1–4.

[40] A. Manjaramkar and R. L. Lokhande, "DEPTA: An efficient technique for web data extraction and alignment," in *2016 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2016*, 2016, pp. 2307–2310.

[41] B. Mehta and M. Narvekar, "DOM tree based approach for Web content extraction," in *2015 International Conference on Communication, Information & Computing Technology (ICCICT)*, 2015.

[42] B. Liu, R. Grossman, and Y. Zhai, "Mining data records in Web pages," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, pp. 601–606.

[43] H. Zhao, W. Meng, Z. Wu, V. Raghavan, and C. Yu, "Fully automatic wrapper generation for search engines," in *International World Wide Web Conference*, 2005, p. 66.

[44] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," *Proc. Elev. Annu. Conf. Comput. Learn. theory - COLT' 98*, pp. 92–100, 1998.

[45] A. H. F. Laender, A. S. Silva, B. a Ribeiro-neto, and J. S. Teixeira, "A Brief Survey of Web Data Extraction Tools," pp. 0–9.

[46] E. Ferrara, P. De Meo, G. Fiumara, and R. Baumgartner, "Web data extraction, applications and techniques: A survey," *Knowledge-Based Syst.*, vol. 70, no. June, pp. 301–323, 2014.

[47] S. H. Muggleton and L. De Raedt, "Inductive Logic Programming: Theory and Methods," *J. Log. Program.*, vol. 19,20, pp. 629–679, 1994.

[48] L. M. Yusuf, M. S. Othman, and J. Salim, "Web classification using extraction and machine learning techniques," *Proc. 2010 Int. Symp. Inf. Technol. - Eng. Technol. ITSim'10*, vol. 2, pp. 765–770, 2010.

[49] T. Ahmad, H. Akhtar, A. Chopra, and M. W. Akhtar, "Satire Detection from Web Documents Using Machine Learning Methods," *2014 Int. Conf. Soft Comput. Mach. Intell.*, pp. 102–105, 2014.

[50] N. G. Ali and N. Omar, "Arabic Keyphrases Extraction Using a Hybrid of Statistical and Machine Learning Methods," *Int. Conf. Inf. Technol. Multimed.*, pp. 281–286, 2014.

[51] M. Mayilvaganan and Sakthivel, "Extraction of Web Information with Implementation of Internet Intelligent Agent System Via Supervised Learning Approach," *Int. J. Comput. Trends Technol.*, vol. 6, no. 1, pp. 42–51, 2013.

[52] A. Talwar and Y. Kumar, "Machine Learning: An artificial intelligence methodology," *Int. J. Eng. Comput. Sci.*, vol. 2, no. 12, pp. 3400–3405, 2013.

[53] S. Aruna and L. V Nandakishore, "An Empirical Comparison of Supervised Learning Algorithms in Disease Detection," *Int. J. Inf. Technol. Converg. Serv.*, vol. 1, no. 4, pp. 81–92, 2011.

[54] R. Amami, D. Ben Ayed, and N. Ellouze, "An Empirical Comparison of SVM and Some Supervised Learning Algorithms for Vowel recognition," *Int. J. Intell. Inf. Process.*, vol. 3, no. 1, pp. 63–70, 2012.

[55] T. Joachims, "Making Large-Scale SVM Learning Practical," 1998.