The 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT) 2018

An Approach of Web Scraping on News Website based on Regular Expression

1st Achmad Maududie
Information System Department
University of Jember
Jember, Indonesia
maududie@unej.ac.id

2nd Windi Eka Yulia Retnani
Informatics Department
University of Jember
Jember, Indonesia
windi.ilkom@unej.ac.id

3rd Muhamat Abdul Rohim
Information System Department
University of Jember
Jember, Indonesia
muhamatabdulrohim@gmail.com

***Abstract*—The high growth of news document emerging a new problem when the news website does not provide downloading service. This paper describes an approach of providing title, publication date, author, clean text article, and URL address of news article from HTML page of three news web-sites, i.e. *Detik*, *Tribunnews*, and *Liputan6* without manually copy and paste process. This approach consists of three steps, i.e.: analyzing news website structure, constructing pattern of Regex and implementing the patterns as a set of rule in web scraping. Based on the experiment, each news web site used their own pattern for article link, article title, article author, and publication date of article. Special for extracting a clean text of news article phase, there were two kinds of pattern i.e.: *content* pattern (for extracting original text article of news) and *filter* pattern (for eliminating non-news elements). In these three-news website, the non-news elements consist of text advertisement, video advertisement, link, image, and script with different pattern for every website. After generated all necessary patterns and implemented these patterns as a set of rules, the web scraping module produced very good results of news article extraction on *Detik* and *Tribunnews* that was presented by recall = 1, precision = 1 and F-Measure = 100% while *Liputan6* had a little bit lower i.e. recall = 0.95, precision = 0.95, and F-Measure = 95%. It is found that this approach is a simple and strait forward way to extract news article which consists of title, publication date, author, news article, and the URL address of news article.**

Keywords—web scraping, regular expression, news article

I. INTRODUCTION

Nowadays, news website has become more and more popular. According to the survey from *Asosiasi Pengusaha Jasa Internet Indonesia* (APJII) 2017, there is 143,26 million internet users in Indonesia which is 58.01% from it uses internet for reading entertainment news, 50.48% sport news, 50.26% social or environmental news, 41.55% religion news, 36.94% political news, and 51.06% health news. Alexa Traffic Rank (ATR) reported that there are three news websites on the top 10 website list of the highest access in Indonesia, i.e., *Detik* (www.detik.com), *Tribunnews* (www.tribunnews.com), and *Liputan6* (www.liputan6.com). Based on our previous observation (6 – 12 June 2018), the average numbers for each day of new news on that three websites were quite big, i.e.: 364 on *Detik*, 374 on *Tribunnews*, and 105 on *Liputan6*.

As a part of our project, i.e. the news topic analysis on newspapers, the high growth of news document emerges a new problem when the news website does not provide API service for downloading these documents. It is not possible to get all of documents by doing copy and paste from the pages of news website according to the number of new documents. Therefore, we tried to develop an approach to solve this problem.

This paper describes an approach of providing clean news article which is grabbed from HTML page without manual copy and paste task from those three news websites. It is based on web scraping technique which was inspired from several previous related works, such as [1], [2] attempted to extract data from HTML table-page, [3] from HTML list-page, while [4]–[6] tried to extract general text (paragraph) from HTML page. However, little bit different with those works, the proposed approach tries to get five elements of news document, i.e.: title, publication date, author, news article, and the URL address of news article from three news web-sites, i.e., *Detik*, *Tribunnews*, and *Liputan6*. In addition, the proposed approach is developed based on regular expression (Regex) to recognize those five elements. The patterns of Regex were derived from the structure of the three HTML news-page.

II. RELATED WORKS

Numerous researchers have been developed a schema-guided approach to get data from HTML pages by representing an HTML document as a tree-like structure which is leaf node represent a data [1]. Vadreu et.al, used presentation regularities and the domain knowledge to develop an automated IE system that automatically transform HTML page into semi-structured hierarchical document with no regards to the domain [2]. Fayzrakhmanov tried to improve the efficiency on information extraction from HTML page based on visual representation [6]. Thamviset and Wongthanavasu introduced a new technique, called Repetitive Subject Pattern, for data extraction based on the hypothesis that each data record has a subject item, there for there will be a repetitive pattern of the subject items and it can be used as a boundary of data record [4]. Thomsen et.al, developed WebSelf, a framework model which models the process of web scraping that consists one selection function, one validation function, two re-induction functions [5]. Han and Tokuda focused on developing a new method (namely a relevance-based analysis) to extract news article content that avoid the page layout analysis and they implemented some HTML tags to indicate the real paragraph of news [6]. Raeymaekers et al. tried to explore unranked tree as a representation of a document and introduced (k, l)-contextual tree languages to extract information from semi-structured document like HTML [7].

Similar with those previous works above, the proposed approach utilizes the existing HTML tags in an as important clue to extract information [8-10]. However, in this approach, we try to construct some Regex patterns as rules to

direct extract the content of HTML news-pages. In this case the content is five elements of news document, i.e. title, publication date, author, news article, and the URL address of news article.

III. METHODS

According to our previous observation, the three-news website have two user-agent for each, i.e. user-agent mobile and user-agent desktop. In order to simplify the extraction process, in this research we selected user-agent mobile as page structure of news websites. Thus, based on this requirement, the proposed approach has three main steps to develop the content of news website extraction tool, i.e.: analyzing news website structure, constructing pattern of Regex and implementing the patterns as a set of rule in web scraping. Quality of the result will be evaluated using precision, recall and F-Measure.

A. News Website Structure Analysis

In general, the news website has two major pages of article [11],[12]. The first major page (usually index page) is a front page of news website which contains a list news title. In addition to title, some of website also add photo and a tiny piece of first paragraph. The most important of this page is recognizing the links to the real news article page (the second major page) which represent the URL addresses of news article. Therefore, in this step we tried to find out which tags that can be used as clues to recognize the links.

The second major page consists of full text of news article. Usually, this page also includes publication date and authors' name. In this page, we tried to find out all tags for four elements of news document, i.e.: titles, author, publication date, and news article. So, in this page we have to get four kinds of tags.

B. Constructing the Pattern of Regex

Based on the clues which was observed in the first step, the next step was generating regex pattern for every element of news document. Consequently, we had to have five type of regex patterns in order to extract title, publication date, author, news article, and the URL address of news article.

C. Evaluation

To ensure the performance, we used precision, recall and F-Measure to measure how well the result was. Recall represents the percentage of news documents that have been extracted over news document on the website while precision represents the percentage of news documents among all extracted documents. To get the overall performance, we used F-Measure which includes Recall and Precision as parameters.

IV. RESULTS AND DISCUSSION

A. The Structure of News Website

The results of the web structure analysis found some differences of tags from the three news websites which was used to present title, publication date, author, news article, and the URL address of news article. The result could be summarizing in Table I.

TABLE I. THE TAGS OF NEWS ELEMENTS FROM *DETIK*, *LIPUTAN* 6, AND *TRIBUNNEWS* WEB PAGES

Element	The Structure of writing HTML pages		
	Detik	***Liputan6***	***Tribunnews***
Link	<a data-category="Subkanal detikNews" data-cation="Indeks" data-label="List Berita" href="Tautan Menuju Halaman Berita" class="list">Judul Berita Yang Dituju	Judul Berita Yang Dituju	<h3 class="blue">Judul Berita Yang Dituju</h3>
Title	Text <h1 class="jdl">Judul Berita</h1> Video <article class="text_area"><h1>Judul Berita Video</h1></article>	<h1 class="article-header__title">Judul Berita</h1>	<h1 class="f32 fno crimson" style="line-height:40px;font-size:34px">Judul Berita</h1>
News Article	Teks Isi Berita Narasi Video Deskripsi Video / Isi Berita	Isi Berita	Isi Berita
Publication Date	Text Tanggal Publikasi Berita Video Nama Penulis/Sumber Berita	Tanggal Publikasi Berita	<time class="grey f13 dip">Tanggal Publikasi Berita</time>
Authors	Text Nama Penulis Video Nama Penulis/Sumber Berita	Nama Penulis/Sumber	Nama Penulis/Sumber

Table I, link of article page in the front page of these three-news website used the same tag, i.e. anchor (<a>) and title of article either used the same tag, i.e. heading1 (<h1>). However, each news website had some different attributes of these tags. Similar with link and title, news article also used the same tag (i.e.) but implemented different attribute. For publication date, each news website employed different

tag, i.e. for *Detik*, for *Liputan6* and <time> for *Tribunnews*. Special for author's element, *Detik* and *Tribunnews* used the same tag (i.e.) while *Liputan6* used . One of interesting facts of the result is *Detik* news-website has a special form of news, i.e. video news. This kind of news used tag <article> for title and tag for news article.

Based on the structure analysis, each article may contain advertisement or other texts that cannot be categorized as news elements. Therefore, in this analysis we also tried to get all non-news elements, such as advertisement, link to other news article, and additional image (photo). The summary of non-news elements analysis result is presented in Table II.

TABLE II. NON-NEWS ELEMENT TAGS

Non-news Element	*Detik*	*Liputan6*	*Tribunnews*
Advertisement	<!--s:parallaxindetail--> Sipan Iklan<!--e:parallaxindetail--> <blockquote>Sisipan Iklan</blockquote>	Sisipan Iklan <blockquote>Sisipan Iklan</blockquote>	Isi Iklan
Other link	<table class="tautansisip">List Tautan</table> 	Isi Tautan Isi Tautan	Baca :judul berita List Tautan
Additional Image	<table align="center" class="pic_article_sisip_table">List Gambar</table> <img src="Sumber gambar">	<img src="Sumber gambar">	<figure>Sisipan Gambar</figure>
Additional script	-	-	<script src="Sumber Script" async="true"></script>
Additional tag	-	-	
Additional video	-	<iframe class="vidio-embed"></iframe>	-

B. Pattern of Regex

After having all important tags, i.e. news elements and non-news elements, the next step was constructing five patterns of regex to produce clean news article. Based on Table I and Table II the patterns of regex for the three-news website could be summarized in Table III, Table IV, Table V, Table VI, Table VII, Table VIII and Table IX.

Table III until Table X present all patterns of regex for news article extraction from web-pages of three news websites. From Table III, IV, V, and VI we could see that each news web site used their own pattern for article link, article title, article author, and publication date of article. Therefore, we created one regex pattern for each element for every news web site as could be seen in these tables. Special for the news article content extraction, we found that news article in each web-page contain not only text of news article, but also five non-news elements, i.e.: text advertisement, video advertisement, image, link to another page, and script (see Table VII until Table X). To get a clean text of news article we created two kinds of pattern that we call *content* pattern (for extracting text article of news) and *filter* pattern (for eliminating non-news elements).

TABLE III. PATTERN OF REGEX FOR ARTICLE LINK OF *DETIK*, *LIPUTAN6*, AND *TRIBUNNEWS*

News Website	Pattern
Detik	<a data-category="Subkanal detikNews" data-action="Indeks" data-label="List Berita" href="(.*?)" class="list">
Liputan6	(.*?)
	<a href="(.*?)" class="article-snippet__title-link"
Tribunnews	<h3 class="blue">(.*?)</h3>
	<a href="(.*?)" title

TABLE IV. PATTERN OF REGEX FOR ARTICLE TITLE OF *DETIK*, *LIPUTAN6*, AND *TRIBUNNEWS*

News Website	Pattern
Detik (Text)	<h1 class=\"jdl\">(.*?)</h1>
Detik (Video)	(.*?)
Liputan6	<h1 class=\"article-header__title\">(.*?)</h1>
Tribunnews	<h1 class=\"f32 fno crimson\" style=\"line-height:40px;font-size:34px\">(.*?)</h1>

TABLE V. PATTERN OF REGEX FOR ARTICLE AUTHOR OF *DETIK*, *LIPUTAN6*, AND *TRIBUNNEWS*

News Website	Pattern
Detik (Text)	<div class=\"author\">(.*?)
Detik (Video)	<span class=\"date\">(.*?) \t\t\t<span
Liputan6	<span class=\"article-header__authors\">(.*?)
	>(.*?)
Detik (Text)	<div class=\"f12 grey mb15\" >(.*?)
	<a href='(.*?)' target='_blank'>

TABLE VI. PATTERN OF REGEX FOR PUBLICATION DATE OF ARTICLE OF *DETIK*, *LIPUTAN6*, AND *TRIBUNNEWS*

News Website	Pattern
Detik (Text)	<div class=\"date\">(.*?)
Detik (Video)	20DETIK \t\t\t<span class=\"date\">(.*?) \t\t
Liputan6	<span class=\"article-header__datetime\">(.*?)
Tribunnews	<time class=\"grey f13 dip\">(.*?)</time>

TABLE VII. PATTERN OF REGEX FOR NEWS VIDEO ARTICLE OF *DETIK*

No	Pattern	Element	Type
1.	<a(.*?)>	Link	Filter
2.	<span class=\"desc\">(.*?)	news content	Content

TABLE VIII. PATTERN OF REGEX FOR NEWS ARTICLE TEXT OF *DETIK*

No	Pattern	Element	Type
1.	<!--s:parallaxindetail-->(.*?)<!--e:parallaxindetail-->	text advertisements	Filter
2.	<table class=\"linksisip\">(.*?)</table>	To locate links	Filter
3.	<table align=\"center\" class=\"pic_artikel_sisip_table\">(.*?)</table>	To locate images	Filter
4.		To locate images	Filter
5.	<div class=\"text_detail detail_area\" id=\"detikdetailtext\">(.*?)	news content	Content
6.	<iframe(.*?)</iframe>	video advertisement	Filter
7.	<a (.*?)>	Link	Filter
8.	<blockquote(.*?)</blockquote>	text advertisement	Filter
9.	<script(.*?)</script>	Script	Filter

TABLE IX. PATTERN OF REGEX FOR NEWS ARTICLE OF *LIPUTAN6*

No	Pattern	Element	Type
1.	<div class=\"seamless-ads__container\">(.*?)	text advertisement	Filter
2.	<div class=\"seamless-ads\"(.*?)	Link	Filter
3.	<div class=\"baca-juga\"(.*?)	link	Filter
4.	<img(.*?)>	image	Filter
5.	<div class=\"article-raw-content\" itemprop=\"description\" data-component-name=\"mobile:article-raw-content\">(.*?)	news content	Content
6.	<a(.*?)>	link	Filter
7.	<blockquote(.*?)100%;\">	text advertisement	Filter
8.	<iframe class=\"vidio-embed \"(.*?)</iframe>	video advertisement	Filter

TABLE X. PATTERN OF REGEX FOR NEWS ARTICLE OF *TRIBUNNEWS*

No.	Pattern	Element	Type
1.	<div class='adspruce-bannerspot'(.*?)	text advertisement	Filter
2.	<span id='innity-in-post'(.*?)	text advertisement	Filter
3.	<div id='div-Inside-MediumRectangle'(.*?)	text advertisement	Filter
4.	<div class=\"txt-article mb20\">(.*?)	new content	Content
5.	Baca: <a(.*?)	link	Filter
6.	(.*?)	link	Filter
7.	<figure(.*?)</figure>	image	Filter
8.	<script(.*?)</script>	script	Filter
9.	<a(.*?)>	link	Filter

As presented in Table VII, *Detik* web-page consist of five non-news elements that had to be filtered, i.e., text advertisement (with two patterns), video advertisement, link (with two patterns), image with (two patterns), and script, while for news video article (Table VIII) only one non-news element, i.e., link. Therefore, we had to generate nine patterns of regex for news text article and one patterns of regex for news video article as presented in Table VII and Table VIII. Similar with *Detik*, *Liputan6* web-page also has several non-news elements, i.e.: text advertisement (two patterns), video advertisement, image, and link to other page (three patterns) as presented in Table IX. So, for *Liputan6* we generated eight patterns of regex. Although had a fewer non-news elements, *Tribunnews* still need some *filter* patterns. Table X shows three patterns for text advertisement, three patterns for link, one for script, and one for image in news text article of *Tribunnews*. Therefore, we generated eight *filter* patterns for this news website.

C. Evaluation

After having all regex pattern (Table III - X), the last step was implementing those patterns of regex in scrapper module and check how well the result was. As we mentioned in methods section, we used recall, precision, and F-measure as the measurement instrument. Based on this measurement, we got scores for each news website as follows.

TABLE XI. RECALL, PRECISION, AND F-MEASURE SCORE

No	News Website	Recall	Precision	F-Measure
1	*Detik*	1	1	100%
2	*Liputan6*	0.95	0.95	95%
3	*Tribunnews*	1	1	100%

Table XI, it is clearly that we got very good result of news article extraction on *Detik* and *Tribunnews*, i.e. recall = 1, precision = 1 and F-Measure = 100%. Although has a little bit less than those two-news web-sites, from the result we could see that *Liputan6* still had a high performance i.e. recall = 0.95, precision = 0.95, and F-Measure = 95%. This table tells us that the propose approach give a good result in providing a clean news article from the three-news website.

V. CONCLUSION

Form the results, we could see that each news website had a unique layout to present their news. Each news website provided one unique HTML element for article link, article title, article author, and publication date of article. Therefore, these news elements could be extracted easily by providing one corresponding regex for each. However, this technique could not be implemented on news article content because it contained some non-news elements in every news website. So, for this news element, we had to provide two kinds of regex, i.e. regex for filtering and regex for extraction.

Based on the evaluations, we got a good result of the proposed approach for these three news websites. *Detik* and *Tribunnews* had recall = 1, precision = 1 and F-Measure = 100% while for *Liputan6* 0.95, 0.95, and 95% respectively. Therefore, it is obvious that the proposed approach could satisfy in providing clean news article from HTML web-page without doing manual copy and paste from the three news websites (*Detik*, *Liputan6*, and *Tribunnews*). This approach is a simple and strait forward to extract news article which consists of title, publication date, author, news article, and the URL address of news article. Thus, this approach could be extended to other news website and could be implemented in the real world.

REFERENCES

[1] X. Meng, H. Lu, H. Gang, and M. Gu, "Data extraction from the web based on pre-defined schema," J. Comput. Sci. Technol., vol. 17, no. 4, pp. 377–388, Jul. 2002.

[2] S. Vadrevu, F. Gelgi, and H. Davulcu, "Information Extraction from Web Pages Using Presentation Regularities and Domain Knowledge," World Wide Web, vol. 10, no. 2, pp. 157–179, May 2007.

[3] W. Thamviset and S. Wongthanavasu, "Information extraction for deep web using repetitive subject pattern," World Wide Web, vol. 17, no. 5, pp. 1109–1139, Sep. 2014.

[4] J. G. Thomsen, E. Ernst, C. Brabrand, and M. Schwartzbach, "WebSelF: A Web Scraping Framework," in Web Engineering, vol. 7387, M. Brambilla, T. Tokuda, and R. Tolksdorf, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 347–361.

[5] H. Han and T. Tokuda, "A Layout-Independent Web News Article Contents Extraction Method Based on Relevance Analysis," in Web Engineering, vol. 5648, M. Gaedke, M. Grossniklaus, and O. Díaz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 453–460.

[6] R. R. Fayzrakhmanov, "Information Extraction from Web Pages Based on Their Visual Representation," in Current Trends in Web Engineering, vol. 7059, A. Harth and N. Koch, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 342–346.

[7] S. Raeymaekers, M. Bruynooghe, and J. Van den Bussche, "Learning (k,l)-contextual tree languages for information extraction from web pages," Mach. Learn., vol. 71, no. 2–3, pp. 155–183, Jun. 2008.

[8] Paul, Subrata, Anirban Mitra, and Swagata Dey. "Issues and Challenges in Web Crawling for Information Extraction." In Bio-Inspired Computing for Information Retrieval Applications, pp. 93-121. IGI Global, 2017.

[9] Pandarge, Sangmesh S., and V. A. Chakkarwar. "Automatic web information extraction and alignment using CTVS technique." In Electronics, Communication and Aerospace Technology (ICECA), 2017 International conference of, vol. 2, pp. 94-99. IEEE, 2017.

[10] Ji, Li, Jiang Guangyu, Xu Aijun, and Wang Yunzhen. "The Automatic Extraction of Web Information Based on Regular Expression." JSW 12, no. 3 (2017): 180-188.

[11] Azir, Mohd Amir Bin Mohd, and Kamsuriah Binti Ahmad. "Wrapper approaches for web data extraction: A review." In Electrical Engineering and Informatics (ICEEI), 2017 6th International Conference on, pp. 1-6. IEEE, 2017.

[12] Rodríguez, Juan M., Hernán D. Merlino, Patricia Pesado, and Ramón García-Martínez. "Evaluation of open information extraction methods using Reuters-21578 database." In Proceedings of the 2nd International Conference on Machine Learning and Soft Computing, pp. 87-92. ACM, 2018.