A Novel Approach to News Archiving from Newswires

Bilkisu Larai Muhammad-Bello[1,2](✉), Mudi Lukman[2], and Mudi Salim[2]

[1] Computer Science and Electrical Engineering G.S.S.T,
Kumamoto University, Kumamoto, Japan
[2] Federal University of Technology Minna, Minna, Niger, Nigeria
bilkisu_bello@futminna.edu.ng, {lukman.mudi,
salim.mudi}@st.futminna.edu.ng

Abstract. A news archive is the core operational tool a media relations team depends on in order to effectively feed a data-hungry organization. An ingrained approach to news archiving in existence is the use of a relational database. As a consequence, integrating search engines that support full-text search is practically impossible due to the strict data schema that is defined in relational database systems. Therefore, there is a need for news archives that support full-text search with relevance ranking of news. In this paper, an approach that supports full-text search is proposed. The process is started by crawling newswire websites for news that are relevant with respect to some predefined keywords and extracting them. Then, they are stored in a data structure known as an inverted-index which supports full-text search, aggregation, and relevance ranking of search results. Search results are ranked and returned to a user in the order of decreasing relevance to the search term. We were able to provide a software solution written in java, the jsoup library for HTML parsing, and an elasticsearch implementation of a search engine. We tested our solution on nine newswires using ten keywords and were able to retrieve a total of 42 relevant news matching seven keywords. The approach proposed in this paper when compared to the manual approach performed better in terms of retrieval speed and accuracy. We conclude that three main components are important in a good digital archive: relevance, extraction, and search. This work is an integration of a good relevance marking technique, an extraction method, and a search engine.

Keywords: Web content extraction · Press review archiving · News extraction from newswires · Search engines as archives

1 Introduction

Newswires syndicate news regarding events, individuals, or organizations which are of interest to specific consumer profiles such as media relation teams, data analysts, scientific researchers, and journalists among others. The periodic avalanche of news generated by these newswires can be used in serving different information needs such as press reviews and news archives. On the one hand, generating a press review does

S. Misra and B. Muhammad-Bello (Eds.): ICTA 2020, CCIS 1350, pp. 546–559, 2021.
https://doi.org/10.1007/978-3-030-69143-1_42

not necessarily require a retrospective of news while on the other hand, news archives must store both past and present news in order to support various tasks such as insight generation through analytics and search engine integration.

Information technology has seen a huge adoption from the digital news industry over the years due to the rapid increase in the amount of information on the internet and how effective, efficient and budget-friendly it supports information delivery [1, 2]. Several web content extraction tools and approaches have been developed in order to successfully and effectively extract the relevant content from webpages. Content extraction from webpages is an intricate task due to the dynamic and ever-evolving nature of webpages and by extension, newswires. Programs known as wrappers specialized for the purpose of extracting relevant content from web sources and mapping them with structures or formats that define a similar or compatible relation are faced with the challenge of recognizing the relevant content from noisy ones [3]. Once extracted, this news can be adapted to support different tasks; news extraction can be applied to generate news highlight sentences that capture the main topic within a news article [4]. Also, the fast and effective extraction of content from webpages could be used to adapt webpages for small screen devices [5] and as proclaimed [6], if we can extract the relevant content of a webpage rapidly, many semantic applications such as search engines can be developed by leveraging this.

In this paper, we present a novel approach to archiving digital news extracted from newswires. Our approach introduces a news archive that extracts news from newswires via tag IDing using predefined keywords. The extracted news is formatted for appropriate storage and indexed in Elasticsearch.[1] We used the jsoup[2] Java[3] library to determine and extract the relevant content. A brief overview of the problem that motivated this work is presented in the next subsection. The novelty of combining a relevance marking approach, content extraction approach, and a search engine as an archive is the crux of this paper and shall be expounded in the following sections.

This paper comprises five main sections. The rest of this paper is organized as follows: a discussion of several existing works on content extraction are presented in Sect. 2; Sect. 3 is the materials and methods section. It presents a discussion of our relevance algorithm and news extraction approach along with details of the search engine. Section 4 presents the results and discussion with a detailed experiment and an evaluation of our approach in comparison with the manual-labor approach. Lastly, we conclude the paper in Sect. 5 with a summary of our work and a brief note on how our approach can be adapted for further work.

1.1 Problem Statement

Several organizations and institutions have specific departments i.e. media relations team whose sole responsibility is to provide members of staff with a summary of work-related news that is of interest to the organization at large (press reviews). This is often achieved through the use of specialized software packages for extracting, formatting,

[1] https://www.elastic.co/.

[2] https://jsoup.org/.

[3] https://www.java.com/en/.

disseminating, and archiving news periodically. This information grows and gradually forms a plethora of news that requires proper maintenance. An ingrained approach to building digital archives in existing works is the use of a relational database that defines a schema. In relational models of databases, all data is technically structured within named relations called tables. Each table is composed of named attributes known as columns. The set of all columns describing a record within a table sums up to what is known as a row. This model was first proposed in 1970 by E. F. Codd in his paper titled A relational model of data for large shared data banks. The concept of normalization was introduced in his paper which in simple terms involves creating relations that have no repeating groups [7].

A major problem surfaces when the need for archived news arises yet they cannot be retrieved even after spending an inordinate amount of time walking through an entire corpus of documents and even putting certain jobs at risk. This is due to the fact that when working with relational databases, search queries are always short in expressive power and features such as context suggestion cannot be leveraged effectively. Therefore, there is a need for a search engine as archive that provides search engine level features such as query autocomplete, context suggestion, analytics support, and provides search results ranked based on their relevance to the search term in real time. In general, features like query suggestion are useful for query term disambiguation and dealing with typographical errors among others [8]. This search engine as archive solution will improve an organizations' business process by saving a sizeable amount of time spent on information retrieval. It will also serve as a fault tolerant solution due to the distributed nature of the underlying search engine implementation presented herein as it allows for replication of multiple copies of data on different machines.

2 Related Work

The goal of content extraction is to produce structured data ready for post-processing, which is relevant in both Web data mining and information retrieval systems [9]. News webpages are made up of varying components such as navigation links, advertisements, social media pointers, and the relevant content of interest as shown in Fig. 1. The speed of information retrieval (IR) can greatly be enhanced through the use of a good document clustering approach. A Modsup-based term frequency and Rider Optimization-based Moth Search Algorithm (Rn-MSA) [10] is presented. It serves as a good document clustering approach that enhances the efficiency of corpus navigation. Content extraction is achieved using the term frequency-inverse document frequency algorithm (TF-IDF) and Wordnet features. Wordnet ontology is used to derive two relations between words: synonyms and hyponyms. Noisy and redundant terms are reduced by removing stop words and stemming inflected words to their root word. Document clustering is performed based on their similarity using the proposed Modsup and Rn-MSA.

An intelligent information system that extracts and archives generic topics from newswires on weekly basis that adapts the term frequency * proportional document frequency algorithm (TF*PDF) is presented in [11]. The TF*PDF of a term is the frequency of the term within a webpage and its frequency within different newswires concurrently.

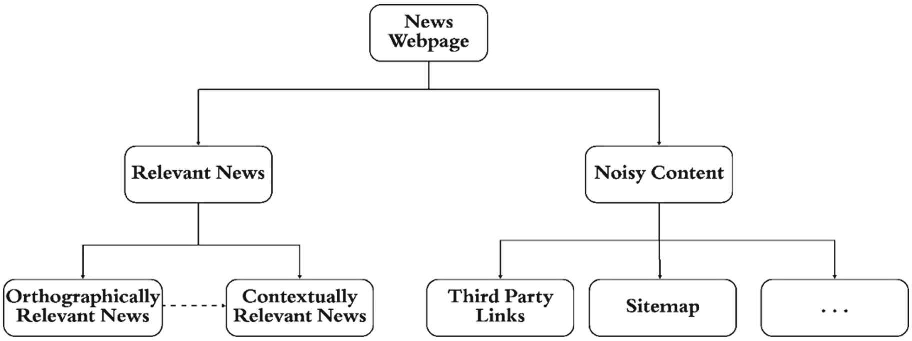


Fig. 1. Classification of contents of a news webpage

The topics containing those terms with high weights are labeled main topics. this approach is not designed to serve press reviews that take keywords into account as such, it is more suited for a popularity ranking use-case as opposed to a relevance ranking one.

Content extraction is being adapted for the extraction of news article narrative [12]. Natural language processing (NLP) methods were employed to enable the analysis and extraction of information from text. These methods were used in extracting individual accounts from news articles by dividing the problem into three steps, namely, named entity recognition, event extraction, and attribute extraction. Named entity extraction was achieved using trained machine learning-based models. A hybrid approach was developed and used to achieve event extraction. Lastly, a dependency parser and Levin's verb classes were used in achieving attribute extraction. CoreEx, a simple heuristic algorithm [5] that extracts the main content from webpages of newswires was developed with an approach that eschewed: the problems associated with structure-dependent approaches e.g. changes in page structure; problems associated with machine learning approaches e.g. re-training as dataset evolves; problems associated with natural language processing (NLP) approaches e.g. their computational expense for large datasets. However, it sometimes excludes the title of a news from the extracted content due to node distance.

A main content extraction algorithm based on node characteristics such as text density and hyperlink density along with neighbor node characteristics was introduced [6]. In abstraction, the algorithm involves six steps: document object model (DOM) generator which is responsible for generating an object model for a webpage, DOM processor does the "housekeeping" of the raw DOM such as extensible hypertext markup language (XHTML) formatting and removal of noisy tags, node fusion uses a similarity algorithm to merge similar nodes, node characteristics analyzer classifies all nodes according to the text density and hyperlink density, node filter filters the noisy nodes, and finally content generator returns the extracted content.

Based on the notion that newer webpages are eschewing the use of structural tags and are adopting an architecture that makes use of stylesheets for structural information, a content extraction technique which was called Content Extraction via Tag Ratios (CETR) [13] was developed in order to keep up content extractors with the aforementioned architectural change. Tag ratio is used to determine content tag by picking the tag with the highest tag ratio from an array of tags based on each tags' content-text value. A template-independent news extraction approach was created in [14] for news aggregators by exploiting the block-oriented structure of webpages. In this approach, a webpage is divided into blocks based on some criteria such as HTML tags. Weights are calculated for each block by considering their textual size and calculating their similarity with the page title. The block with the highest weight is selected as the news block. The use of a similarity model in this approach makes it an outlier from existing block-based approaches. This is because it increases the accuracy of news block detection. It does not rely on textual size only, but also on title-block similarity due to the possibility of a noisy block having a higher weight than a news block with fewer text. Multithreading and recursion were applied in [15] to design a highly scalable bytecode-based java archive search engine to manage the rapid growth in data size.

The application of text ranking cuts across different application areas, primarily used in IR; text ranking is being used also for NLP. In [16], an overview of a framework for text ranking using transformers is presented. Transformers are a kind of neural network architecture for text ranking. The exact type of transformer employed is a bidirectional encoder representation from transformers (BERT) by Google [17].

3 Materials and Methods

3.1 Relevance Marking

In our approach, a news can be either orthographically relevant, contextually relevant or both. If the set of tags that qualify the news as relevant contains only acronyms and abbreviations, then the news is simply classified as orthographically relevant e.g. “Sec.” could mean “Securities and Exchange Commission” or just an abbreviation of “Secondary”. Finally, if the set of tags that qualify the news as relevant contains at least one jargon, noun, or phrase, it is classified as both orthographically and contextually relevant as shown by the dashed arrow in Fig. 1. Relevant news articles can become less relevant as more documents are added to a search engine over time. This can be due to poor-quality content being added and the daily refresh interval of what users view as up-to-date [18]. The relevance scoring function for documents relevant to a query cannot always be ran for all documents in a large scale search engine due to the impact of the corpuses size on computational cost [19]. Both arguments have a huge impact on news relevance marking. A news webpage always contains both the relevant content i.e. title and body and the noisy content which includes elements such as external links, sitemaps, advertisements, comment blocks, etc. Advertisements can be removed from web pages during the parsing process using the common and efficient approach of analyzing HTML tag attributes. Our focus is on the relevant content so we restrict the discussion of noisy content to what has been mentioned earlier. Tags or keywords which represent topic of interest are used to determine whether a news is relevant or not. As described, there exists a dichotomy between semantic matching and relevance matching [20]. On the one hand, sematic matching involves identifying the language level meaning of text and the language level relationship existing between two homogenous texts. On the other hand, relevance matching involves determining whether a document is relevant to a given query term.

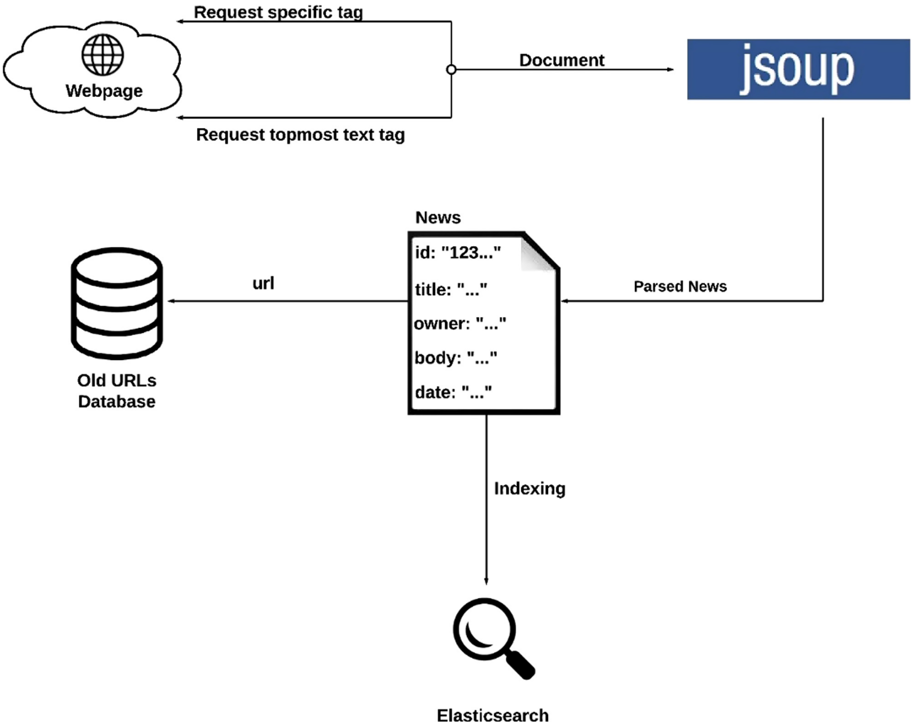


Fig. 2. Key stages in our news archive system

3.2 News Extraction

In order to accomplish an effective extraction of the relevant news, rather than reinventing the wheel, our approach employs the jsoup (version 1.12.1) extraction library. We used a selector-syntax to mark specific HTML tags of familiar newswire webpages i.e. < `div.entry-content` > tag was used as with the following: ‘String body = newsContent.select(“div.entry-content”).first().text();’ while for unfamiliar newswires, we simply select the topmost text tag from the DOM tree of its webpage. The extracted news is then associated with some attributes to reference it in memory i.e. a unique ID to uniquely identify each instance of a news, a title attribute that stores the news title, an owner attribute that stores the uniform resource locator (URL) of the newswire from which the news was extracted, a body attribute that stores the actual text of the extracted news, and a date attribute that stores a timestamp for the extracted news for future reference or to enable tasks like sorting of news in chronological order. The URL of the extracted news is added to a list of old URLs backed by a permanent storage location in order to avoid duplication of effort in subsequent extractions by skipping all URLs in the list of old URLs. The extraction process is completed by sending the extracted news to Elasticsearch for indexing as shown in Fig. 2.

3.3 Indexing and Search

Google ranks pages based on their popularity among other pages i.e. how many other pages link to it. As at the time of this writing, Google, Bing, Baidu, Naver, and Yahoo among others are the most used search systems across the globe [21]. An inverted index is the core of all modern-day search engines. An inverted index is a data structure that creates a mapping between unique tokenized terms and their associated information such as to which document do they belong and at what position in the document do they appear. This data structure is a popular choice with search engine designers due to its speedy provision of search results. Our approach uses Elasticsearch as the archive for all extracted news. Even though it is almost an indispensable choice of search engine implementation for our work, the key motivation to why we preferred and used it is its speed and distributed nature. These powerful characteristics it has can be leveraged in the future to scale a platform to very large number of server machines and data flow with minimal effect on performance [22, 23]. User expectation from search engines with regards to response time has greatly increased, owing to factors like increase in network speed and user shrewdness [24]. As the reader may have guessed, Elasticsearch also relies on the use of an inverted index as the data structure for storing input data. A sample text and its corresponding inverted index is presented below. (Table 1)

"Take care of five before five"

Assuming the document name is 1;

Table 1. The inverted index built for the sample text

Term	Document:Position
Take	1:1
care	1:2
of	1:3
five	1:4, 1:6
before	1:5

Before the indexed news can be searchable, a pseudo-schema (lazily enforced) has to be defined to map all the attributes of the news i.e. id, title, body, owner, and date to a specific data type. For this, we map the id, title, body, and owner attributes to a string data type and map the date attribute to a date data type. Our software implementation[4] communicated with Elasticsearch through Elasticsearch's Java high level REST client 6.3 library and data is being sent and received in JavaScript object notation (JSON) format.

[4] https://github.com/MudiLukman/news-archive.

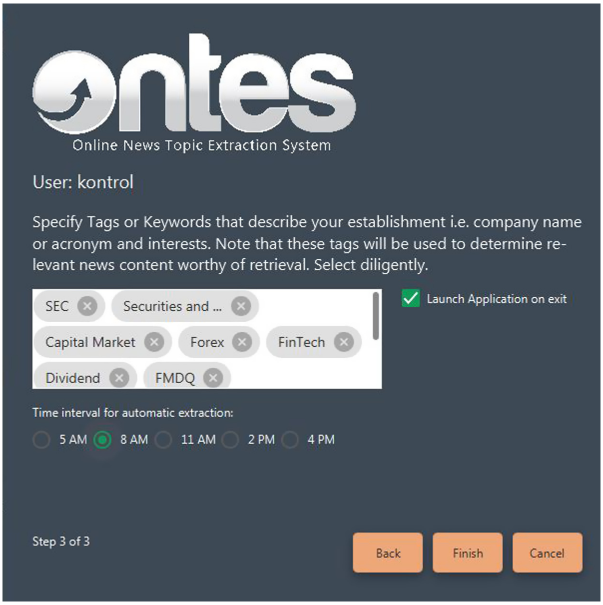
4 Results and Discussion

4.1 Implementation

Our approach was implemented in Java using the IntelliJ[5] version 2017.2.2 integrated development environment (IDE) bundled with the Java Development Kit version 8 (JDK 8). The experiment was carried out on a computer with an Intel Pentium B960@2.20 GHz CPU with 5 GB of main memory on a 64-bit windows 7 ultimate environment. The primary reason Java was chosen as the language of choice is its portability. Java programs are portable in that, they allow programmers write and deploy the same codebase on different types of machine architecture in a technique that employs the use of a bridge language known as bytecode. These bytecode instructions can run on any machine that has the Java Virtual Machine (JVM) installed. Although, the notion of universal portability of Java is not entirely correct due to recent platform-dependent evolution of some APIs as is the case with some classes in the New I/O (NIO) API in recent versions of Java [25, 26]. Elasticsearch 6.3.0 configured to use the *multi match* full-text search query which allows us search multiple fields in a single pass, say *title* and *body* was used. The default analyzer i.e. standard analyzer was not altered in any form. We briefly describe the visual interface of our implementation below.

User Interface

The first point of interaction between the system and a user is the login page which leads to the window depicted in Fig. 3. This is the final stage in the initial installation process



ontes

Online News Topic Extraction System

User: kontrol

Specify Tags or Keywords that describe your establishment i.e. company name or acronym and interests. Note that these tags will be used to determine relevant news content worthy of retrieval. Select diligently.

SEC ⊗ Securities and ... ⊗ Capital Market ⊗ Forex ⊗ FinTech ⊗ Dividend ⊗ FMDQ ⊗

☑ Launch Application on exit

Time interval for automatic extraction:

○ 5 AM ● 8 AM ○ 11 AM ○ 2 PM ○ 4 PM

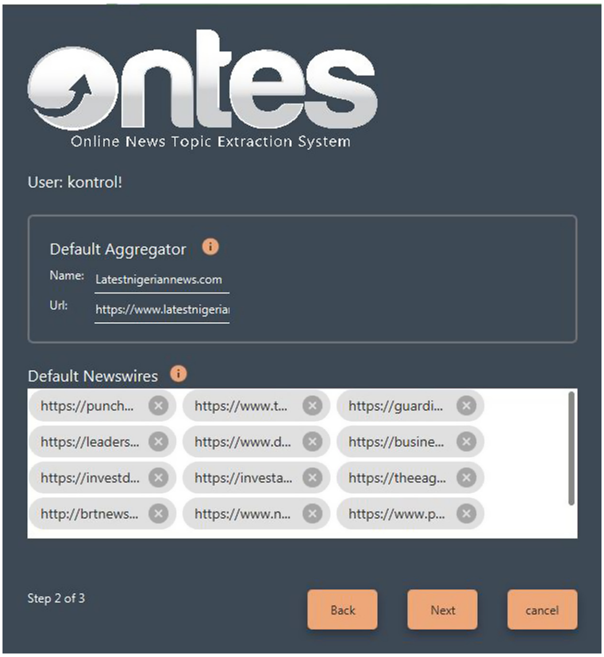
Step 3 of 3

Fig. 3. Specifying keywords and tags at the installation stage

[5] https://www.jetbrains.com/idea/.

where a user is prompted to sets the various tags (which includes topics, acronyms, words, initials and trending topics) or keywords that best represents the interests of the organization. The default time interval for automatic news content extraction is also set as depicted in Fig. 3.

Figure 4 depicts the window where information regarding data sources are collected. The system requires a default news aggregator source to fetch daily news content to be displayed. A minimum of one newswire source is required to create a pathway for data that is required to be fetched from the internet. The data is then parsed, and indexed in the search server.



User: kontrol!

Default Aggregator

Name: Latestnigeriannews.com

Url: https://www.latestnigeriai

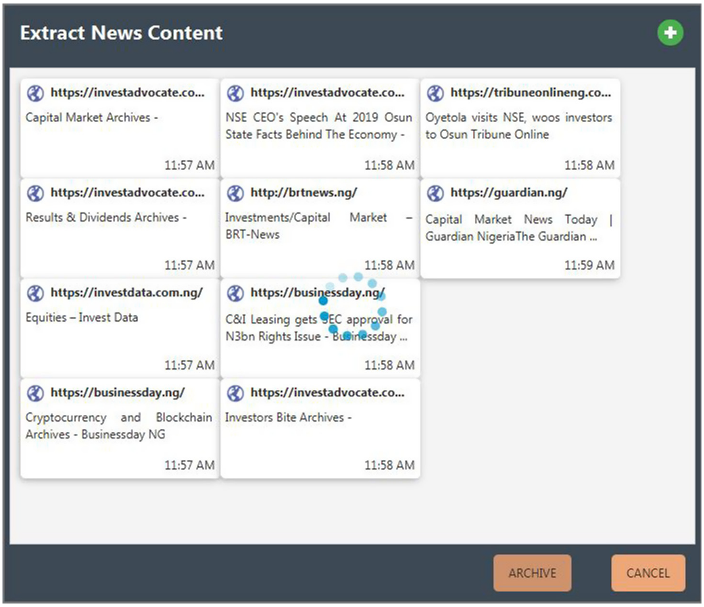
Default Newswires

https://punch... https://www.t... https://guardi...
https://leaders... https://www.d... https://busine...
https://investd... https://investa... https://theeag...
http://brtnews... https://www.n... https://www.p...

Step 2 of 3

Fig. 4. Collecting data sources and specifying a Default Aggregator

Figure 5 shows the core fundamental operations of news archiving system at work. The operational stages include: web crawling, content extraction, document parsing, and archiving in an inverted index. The content area depicted in Fig. 5 continues to display all news contents as they are being extracted from the web and considered relevant. A topic is considered relevant if it contains at least one of the tags or default keywords specified during the installation stage. Clicking on the Archive button at the bottom-right corner of the window saves all the extracted news contents in an inverted index while a click on the plus icon allows the user to input web links to news documents considered to be index-worthy.



https://investadvocate.co...
Capital Market Archives -
11:57 AM

https://investadvocate.co...
NSE CEO's Speech At 2019 Osun State Facts Behind The Economy -
11:58 AM

https://tribuneonlineng.co...
Oyetola visits NSE, woos investors to Osun Tribune Online
11:58 AM

https://investadvocate.co...
Results & Dividends Archives -
11:57 AM

http://brtnews.ng/
Investments/Capital Market – BRT-News
11:58 AM

https://guardian.ng/
Capital Market News Today | Guardian NigeriaThe Guardian ...
11:59 AM

https://investdata.com.ng/
Equities – Invest Data
11:57 AM

https://businessday.ng/
C&I Leasing gets SEC approval for N3bn Rights Issue - Businessday ...
11:58 AM

https://businessday.ng/
Cryptocurrency and Blockchain Archives - Businessday NG
11:57 AM

https://investadvocate.co...
Investors Bite Archives -
11:58 AM

Fig. 5. Content area of the news archiving system

4.2 Experimental Data Source

We used 3,018 webpages from 9 newswires and 10 keywords to test the effectiveness of our approach. The basic information of the experimental newswires is shown in Table 2.

Table 2. A list of the experimental newswires used

Newswire	URL	Number of traversable links
BRT-News	http://brtnews.ng/	332
Invest Data	https://investdata.com.ng/	106
Invest Advocate	https://investadvocate.com.ng/	201
Business Day	https://businessday.ng/	473
Punch	https://punchng.com/	155
The Nation	https://thenationonlineng.net/	731
Daily Trust	https://www.dailytrust.com.ng/	218
Independent	https://www.independent.ng/	499
New Telegraph	https://www.newtelegraphng.com/	303
Total	**9**	**3,018**

4.3 Experimental Results

The results of the approach to news archives we proposed is shown in Table 3. This result includes the news count for each matched keyword per newswire. A total of 42 news was extracted and archived from 3,018 webpages matching 7 of 10 keywords. The experimental results drawn from our approach proves that this approach can be used in practical environments for news extraction due to its accuracy

Table 3. Results for news extracted for each keyword per newswire

Keywords	BRT-News	Business Day	Invest Advocate	Invest Data	Punch	The Nation	Daily Trust	Independent	New Telegraph
Capital Market		1	4	1		1			
Cryptocurrency		1							
Dividends		1	3						
Equities			1			2			
FinTech									
Forex									
Investors			2	1					
SEC	1	3	7	2	1	3	3		3
Securities and Exchange Commission								1	
Stock Market									
Total	**1**	**6**	**17**	**4**	**1**	**6**	**3**	**1**	**3**

An important, if not mandatory characteristic of an algorithm is that it is effective in solving the problem for which it was designed [27]. However, some newswires refused to accept incoming connections and a request-timeout error was returned. So, no news was extracted from these newswires.

As shown in Table 3, seven news webpages matching the keyword "**Capital Market**" were extracted from four different newswires; one each from *Bussinessday*, *InvestData*, and The Nation while four from *InvestAdvocate.* This includes news that contains the keyword within either its title, body, or both. one webpage matching the keyword "**cryptocurrency**" was marked as relevant and was extracted from *Businessday.* Four webpages from two newswires (one from *Businessday* and three from *InvestAdvocate*) matching the keyword "**Dividends**" were marked as relevant and were consequently extracted. Three webpages were found matching the keyword "**Equities**" from two sources; one from *InvestAdvocate* and two from The Nation. a total of three webpages matching the keyword "**Investors**" were extracted from two sources; two from *InvestAdvocate* and one from *InvestData.* A total of twenty-three news webpages matching the keyword "**SEC**" were found and extracted from eight sources; one each from BRT-News and Punch, three each from *Businessday*, The Nation, Daily Trust, and New Telegraph, two from *InvestData*, and seven from *InvestAdvocate.* finally, one news webpage matching the keyword "**Securities and Exchange Commission**" was found and extracted from independent.

4.4 Evaluation

To measure the performance of our approach in terms of its effectiveness, we used the following criterion to evaluate our experimental result. To accomplish this, we first carried out news extraction manually. Our approach is used afterwards. We assign a score of 1 for every correct extraction made on each side. In the end, the manual approach had a total of 34 points, while our approach had a total of 42 points.

To measure the efficiency of our approach, we also recorded the total execution time for news extraction using our approach in comparison with that of the manual approach. Our program had an execution time of 270 s while the manual process took 3,360 s to complete. The results obtained above shows that our approach outperforms the manual approach to news extraction.

5 Conclusion

News extraction and archiving is important in businesses and to organizations. We presented a novel approach that combines an extraction method for news webpages with a search engine as an archive to support information retrieval with a nice architecture, providing an analytics platform, eliminating the likelihood of errors resulting from human intervention in news extraction and archiving, a profound usage of multithreading and parallel computing techniques, avoiding duplication of effort by skipping archived news during extraction, and consequently affecting business processes positively. Multithreading and recursion were applied to design a highly scalable bytecode based java archive search engine to manage the rapid growth in data size. We therefore recommend that the approach to news archives presented herein be put into practical use by organizations and news aggregators that work with real time news wherever applicable.

In further works, our approach can be extended to support real time dispersal of extracted news to subscribed clients at periodic intervals.

References

1. García, R., Perdrix, F., Gil, R.M.: Ontological infrastructure for a semantic newspaper. In: Proceedings 15th World Wide Web Conference on Semantic Web Annotations for Multimedia (SWAMM), pp. 1–12 (2006)
2. Crescenzi, V., Mecca, G., Merialdo, P.: RoadRunner: towards automatic data extraction from large web sites. In: VLDB 2001 - Proceedings of 27th International Conference on Very Large Data Bases, pp. 109–118 (2001)
3. Laender, A.H.F. Ribeiro-Neto, B.A., da Silver, A.S., Teixerira, J.S.: A brief survey of web data extraction tools. ACM SIGMOD Rec. **31**(2), p. 84 (2002). https://doi.org/10.1145/565117.565137
4. Wong, K.-F. et al.: Utilizing microblogs for automatic news highlights extraction. In: Series on Language Processing, Pattern Recognition, and Intelligent Systems. Social Media Content Analysis, pp. 277–296 (2017). https://doi.org/10.1142/9789813223615_0019
5. Prasad, J., Paepcke, A.: {CoreEx}: content extraction from online news articles. In: CIKM 2008 Proceeding 17th ACM Conference on Information and Knowledge. Management. pp. 1391–1392 (2008). https://dl.acm.org/doi/10.1145/1458082.1458295

6. Liu, Q., Shao, M., Wu, L., Zhao, G.: Main content extraction from web pages based on node characteristics. J. Comput. Sci. Eng. **11**(2), 39–48 (2017). https://doi.org/10.5626/JCSE.2017.11.2.39
7. Connolly, T., Begg, C.: Database Systems: A practical Approach to Design, Implementation and Management (Sixth Edition). Pearson, Boston (2014)
8. Dehghani, M., Rothe, S, Alfonseca, P., Fleury, P.: Learning to attend, copy, and generate for session-based query suggestion. In: Proceedings of the 2017 ACM International Conference on Information and Knowledge Management, pp. 1747–1756 (2017). https://doi.org/10.1145/3132847.3133010
9. Negm, N., ElKafrawy, P., Salem, A.B.: A survey of web information extraction tools. Int. J. Comput. Appl. **43**(7), 19–27 (2012). https://doi.org/10.5120/6115-8296
10. Yarlagadda, M., Gangadhara Rao, K., Srikrishna, A.: Frequent itemset-based feature selection and Rider Moth Search Algorithm for document clustering. J. King Saud Univ. Comput. Inf. Sci. (2019). https://doi.org/10.1016/j.jksuci.2019.09.002
11. Bun, K.K., Ishizuka, M.: Topic extraction from news archive using TF*PDF algorithm In: Proceedings of the Third International Conference on Web Information Systems Engineering WISE 2002, IEEE, pp. 73–82 (2002). https://doi.org/10.1109/wise.2002.1181645
12. Zhang, H., Boons, F., Batista-Navarro, R.: Whose story is it anyway? automatic extraction of accounts from news articles. Inf. Process. Manag. **56**, 1837–1848 (2019). https://doi.org/10.1016/j.ipm.2019.02.012
13. Weninger, T., Hsu, W., Han, J.: CETR - content extraction via tag ratios. In: Proceedings of the 19th International Conference on the World Wide Web, WWW 2010. pp. 971–980 (2010). https://doi.org/10.1145/1772690.1772789
14. Yenicag, A., İnternet, H., İçin, S., İçerik, S-B., Yöntemi C.: A Template-Independent Content Extraction Approach for News (2012)
15. Karnalim, O.: Improving scalability of java archive search engine through recursion conversion and multithreading. CommIT (Commun. Inf. Technol.) J. **10**, 15–26 (2016). https://doi.org/10.21512/commit.v10i1.832
16. Lin, J., Nogueira, R., Yates, A.: Pretrained Transformers for Text Ranking: BERT and Beyond (2020)
17. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL HLT 2019 – 2019 Conference North American Chapter of the Association for Computational Linguistics Human Language Technologies - Proceeings Conference volume 1, pp. 4171–4186 (2019)
18. Croft, W.B., Metzler, D., Strohman, T.: Search Engines Information Retrieval in Practice. Pearson, Prentice Hall (2015)
19. Nakamura, T.A., Calias, P.H., de Castro Reis, D., Lemos, A.P.: An anatomy for neural search engines. Inf. Process. Manag. **480**, 339–353 (2019). https://doi.org/10.1016/j.ins.2018.12.041
20. Guo, J., Fan, Y., Ai, Q., Croft, W.B.: A deep relevance matching model for Ad-hoc retrieval. In: Proceedings of the 25th ACM International Conference on Information and Knowledge Management, pp. 55–64 (2016). https://doi.org/10.1145/2983323.2983769
21. Liu, B.: Web data mining: exploring hyperlinks, contents, and usage data. In: Carey, M.J. and Ceri, S. Data Centric Systems and Applications. (Second Edition). Springer, Berlin (2015). https://doi.org/10.1007/978-3-642-19460-3
22. Dixit, B., Kuć, R., Rogoziński, M., Chhajed, S.: Elasticsearch A Complete Guide. Packt Publishing, Sebastopol (2017)
23. Gormley, C., Tong, Z.: Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine. O'Reilly Media, Sebastopol (2015)
24. Brutlag, J.D. Hutchinson, H., Stone, M.: User preference and search engine latency. In: JSM Proceedings, Quality and Productivity Research Conference, American Statistical Association, vol. 12, pp. 1–13 (2008)

25. Schildt, H.: Java The Complete Reference (Eleventh Edition). McGraw-Hill, New York (2018)
26. Liang, Y.D.: Introduction to Java Programming Comprehensive Version. Pearson, Prentice Hall (2016)
27. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein C..: Introduction to Algorithms. MIT Press (3^{rd} Edition), Cambridge (2009)