Development of Focused Crawlers for Building Large Punjabi News Corpus

Gurjot Singh Mahi* **& Amandeep Verma**

Department of Computer Science, Punjabi University
Patiala, Punjab, 147002 India
*E-mail: gurjotmahi28@gmail.com

Abstract. Web crawlers are as old as the Internet and are most commonly used by search engines to visit websites and index them into repositories. They are not limited to search engines but are also widely utilized to build corpora in different domains and languages. This study developed a focused set of web crawlers for three Punjabi news websites. The web crawlers were developed to extract quality text articles and add them to a local repository to be used in further research. The crawlers were implemented using the Python programming language and were utilized to construct a corpus of more than 134,000 news articles in nine different news genres. The crawler code and extracted corpora were made publicly available to the scientific community for research purposes.

Keywords: *corpus; crawler; NLP; Punjabi language; scraper; text extraction; text processing.*

1 Introduction

Deep learning methodologies have changed the paradigm of computer science research in recent times. These methodologies are capable of discovering intricate structures hidden in extensive corpora [1]. A corpus is a large structured collection of texts used by researchers to make statistical inferences or to develop computer applications. Natural Language Processing (NLP) is a subfield of computer science and linguistics, in which computers are programmed to process and analyze natural language data. The majority of these NLP applications are trained on a corpus or data provided during their development. Researchers, academicians and scientists use publicly available large corpora for solving complex multilingual NLP problems, like speech processing, sentiment analysis, machine translation, and text generation.

The World Wide Web is the single largest source of data or information commonly utilized to fuel large-scale biologically inspired neural algorithms and systems. Online resources on the Web are recurrently used to build large corpora, which are further used to develop multifaceted computer applications relying on the information learned from the corpora. For example, Ref. [1] mentions that word vector learning works very well when the data comes from very large corpora.

Received April 14th, 2021, Revised July 12th, 2021, Accepted for publication September 8th, 2021.
 ISSN: 2337-5787, DOI: 10.5614/itbj.ict.res.appl.2021.15.3.1

Manual development of a large corpus requires intensive resources, time, cost, and manual work, which is usually not feasible in a typical research laboratory environment. To overcome this problem crawlers are programmed to automatically extract large amounts of text. A web crawler is a data acquisition tool used by search engines [2]. Crawling websites is a recursive process.

A typical crawler works in the following manner: a) the crawler is given a list of seed URLs to start the crawling process; b) the crawler sends a URL request to each webserver to check if the URL is active; c) it fetches the webpage through an HTTP client; d) the fetched webpage is parsed and the needed information is extracted from the parsed webpage; and e) the extracted information is saved in a local repository. This crawling process is continuously repeated until the crawler has visited all the URLs in the list. The resulting large corpora are utilized to develop specialized systems, but also aid in increasing system accuracy of NLP applications. For example, when a training corpus is increased from 500,000 to 3 million words, the accuracy of the word prediction system is enhanced to 54.4% from 50.2% [3].

Some work has been done on developing corpora using crawlers for NLP applications in the Indo-Aryan and Dravidian language groups of India. A part of speech (PoS) tagged Bengali news corpus was developed in [4] using a web crawler. Ref. [5] developed a Hindi-English parallel corpus for statistical machine translations system primarily using web crawling. However, when it comes to other Indo-Aryan languages such as Punjabi, to the best of our knowledge no open-source crawler for corpus creation has been developed yet. Several researchers have mentioned the development of Punjabi text corpora and datasets through crawlers but have not released any source code for research purposes.

In this paper, the source code[1] for three focused open-source crawlers are proposed. These were used to extract corpora[2] from three Punjabi news websites for automated corpus development. The proposed method can be used by any researcher who wants to develop a Punjabi news corpus for natural language processing tasks or any other relevant field of study.

The rest of this paper is divided into five sections. The background of this research and the motivation behind the development of the mentioned crawlers is described in Section 2. The architecture of the crawlers is described in detail in Section 3. The results and future work are discussed in Section 4. The paper ends with the conclusion in Section 5.

[1] https://github.com/GurjotSinghMahi/punjabi_news_website_crawlers

[2] https://drive.google.com/drive/folders/1bB3hmTr4COMMUijEx8BAcMYCfsvh7xJW?usp=sharing

2 Background and Motivation

Web crawlers, or scrappers, are specialized automated computer programs that traverse a website schema as a part of search engines that maintain an index of the web pages. The development of automated crawlers helps in the automatic extraction of large amounts of data, which can be of great importance for the research community. Large data gathering is costly and resource-intensive, requiring hours of manual work by the researcher. For this reason web crawlers have been developed.

Crawlers are primarily designed for efficient data extraction, while being able to handle obstacles like varying load speed, IP blocking, URL errors, timeout errors, server bots, and many other difficulties. All these issues need to be addressed while designing or developing a website crawler.

Crawlers such as RBSE Spider [6], Mercator [7], UbiCrawler [8], BlogForever crawler [9], GitcProc [10] were developed for diverse domains such as search engines, blogs, GitHub commits, etc. Several studies have been done on the development of focused crawlers for corpus generation in different domains, for example [11-15]. Ref. [11] developed a web crawler for building corpora specific to the CAD domain.

The study reported in [12], which majorly inspired the study reported in this article, developed a crawler for obtaining recruitment website data for corpus development. Other than this, crawlers have also been developed for social media data extraction.

A focused crawler called TwitterEcho [13] was developed to extract Portuguese language tweets for research purposes from Twitter. Ref. [14] developed a Facebook website crawler to retrieve comments from Facebook posts, which was able to collect 7,567 comments. The Automatic Social Emotion Detection System (ASEDS) was engineered in [15] using 3 million posts from 64,000 Facebook pages of different domains, obtained by developing a scalable crawler. In the same line, Ref. [16] used a Heritrix open-source crawler to crawl 6,900 URLs for building an NoWaC corpus for the Bokmål Norwegian language, containing 700 million tokens.

India is a multilingual country and the Eighth Schedule to the Indian Constitution lists 22 national languages [17]. The Indo-Aryan language (Punjabi) has a speaker base of more than 100 million speakers across the globe. Previous works like [18] and [19] outline the development of Punjabi corpora in distinct domains, but none have released a corpus for future use. Also, much of the development of different corpora has been done manually. The unavailability of a corpus for the Punjabi

language inspired us to develop automated crawlers for corpus creation. Our contribution to this topic is as follows:

1. The overall architecture of a crawler for creating news corpora for a low resource language (Punjabi) is described in detail, which automatically extracts news articles from Punjabi news websites while retaining metainformation embedded in news webpages.
2. Overall, 134,000 news articles in nine different news domains were downloaded, making it one of the most extensive corpora for the Punjabi language.
3. For the first time, an open-source focused crawler has been developed and made freely available for three significant news websites in the Punjabi language and made publicly available for this low resource language.
4. Overall, this research is of great importance for building natural language processing applications for the low resource language Punjabi.

3 Crawler Development

This section discusses the selection of the target websites and the system architecture of the crawlers.

3.1 Selection of Websites

The first step was to select quality sources of text, in our case, news websites publishing quality new articles. Three Punjabi news websites, punjabitribuneonline.com, punjabijagran.com, and jagbani.punjabkesari.in, were selected to develop the focused crawlers for corpus creation. The mentioned websites are the most frequently visited websites for reading news in Punjabi. Another main reason for selecting the mentioned websites is the similarity in the publishing structure of these websites. Each news article on the chosen websites contains article title, article text, and other metainformation, such as publishing date, month, and year, which were of great importance for our research study.

Figure 1 illustrates the resemblance between the three Punjabi websites. Our task was to develop an automated set of crawlers, which could extract quality text articles from these websites while retaining the metainformation of the extracted articles.

(a) Punjabi Tribune

(b) Jagbani

(c) Punjabi Jagran

Figure 1 The structure of the three selected Punjabi websites, where (a) is the article title, (b) illustrates the article meta-information, and (c) is the published article text.

3.2 Software Architecture

Although the metainformation structure embedded in the selected websites was the same, each website was constructed using different HTML attribute naming conventions, which makes it hard for a single crawler to crawl all websites. This anomaly forced us to create three discrete crawlers for each selected website with the same system architecture.

The crawlers were developed using the Python programming language [20], utilising the Urllib [21] and BeautifulSoup [22] modules. Figure 2 describes the crawling architecture adopted for the development of the crawlers. The architecture is divided into two phases:

1. Page URL extraction
2. News article extraction

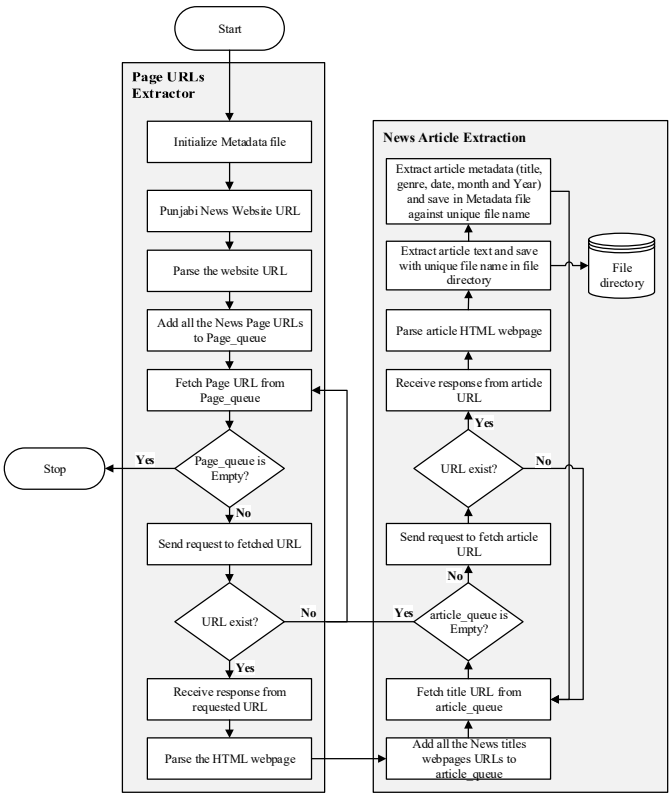


Figure 2 The system architecture of the Punjabi news website crawlers.

3.2.1 Page URL Extraction

This phase of the crawler is responsible for extracting all the URLs containing HTML pages, which includes creating a list of published articles, achieved by maintaining the `Page Queue` file. This phase starts by initializing the Metadata file, which will be used for saving the metainformation in the next step. Next, the `Page Queue` links are extracted using the FIFO (First In – First Out) scheduling scheme.

Each HTML page URL is fetched from the `Page Queue` and an HTTP request is sent to the fetched page. If the URL exists and a response is received from the news website server, the HTML page is parsed and sent for the next phase of text extraction, else the next URL is fetched from the `Page Queue`. This process is iterated until the `Page Queue` is empty and the crawling process is stopped.

3.2.2 News Article Extraction

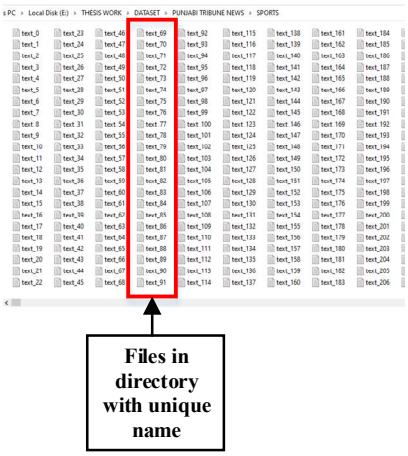
In this phase, all the article links embedded in the parsed HTML page that were visited in the previous stage are extracted and added to the `Article Queue`. The URLs that link to published articles are fetched from the `Article Queue` and an HTTP request is sent to the server. If an article URL does not exist, the next URL is fetched from the `Article Queue`, else the article URL response is received from the server. The HTML webpage is parsed using curated rules based on the different HTML naming conventions for each website.

The article text is extracted from the parsed webpage and saved in text (.txt) file format using Unicode (UTF-8) encoding, making the extracted text operating-system independent. The text file is transferred to a file repository with unique nomenclature and the article meta-information, i.e. title, date, month, and year, is extracted from the parsed article. The extracted metainformation is saved in the Metadata file that was initialized in the previous phase (page URL extraction) for each unique file name. This process is iterated until the `Article Queue` is empty. Once the `Article Queue` is empty, the process jumps back to the page URL extraction phase. The following page URL is fetched, and phase 2 starts again.

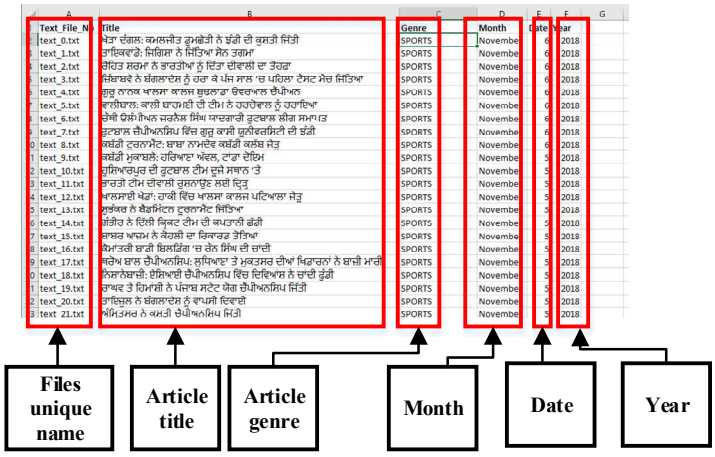
4 Results and Future Work

Along with the growth of the Internet, the digital news industry is also booming. Every day an enormous amount of news articles are published on Punjabi news websites. The developed crawlers can extract this growing amount of data. They have pulled more than 134,000 articles in nine genres (Sports, Special Page, Regional, Editor Page, Kids, International, Entertainment, Business, and Special

Editor) of news. The metainformation is retained in a file while extracting the news articles, containing the article title, genre, date, month, and year, for each unique file name. The extracted corpus contains more than 6 million word tokens and more than 500,000 sentences. The authors utilized the developed corpus to develop the first sentence completion system for the Punjabi language. An example of the files extracted and the metadata file containing the metainformation is shown in Figure 3.



(a) File directory containing text files with unique file names



(b) Metadata file containing metainformation about text articles with unique file names

Figure 3 Structure of the file directory and the file with metadata collected during website crawling.

Future work includes the extension of the performed work to other Punjabi news websites. We are also aware that some of the articles may not contain the quality text needed, so a more sophisticated crawler architecture will be developed by adding quality weighing measures to the system to assess the quality of each text before extracting. Another limitation of the proposed crawlers is that they only extract data from the three mentioned websites.

5 Conclusion

The present paper presents a novel architecture for Punjabi news crawlers. The crawlers' primary goal is to extract news articles from three prominent Punjabi news websites. We utilized the mentioned crawlers to construct a corpus of more than 134,000 news articles in nine different news genres for our own research purposes. To the best of our knowledge, these are the first open-source crawlers developed for crawling news websites.

The created corpora are provided to the scientific community for research purposes via a weblink. The developed corpora can be utilized to create classification systems, predication applications, sentence completion systems and other NLP applications for the Punjabi language. We hope that the presented tools will contribute to the elevation of a low resource Indo-Aryan language such as Punjabi.

References

[1] LeCun, Y., Bengio, Y. & Hinton, G., *Deep Learning*, Nature, **521**(7553), pp. 436-444, 2015.

[2] Kumar, M., Bhatia, R., & Rattan, D., *A Survey of Web Crawlers for Information Retrieval*, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, **7**(6), pp. 1-45, 2017.

[3] Lesher, G.W. & Sanelli, C., *A Web-based System for Autonomous Text Corpus Generation*, in Proceedings of ISSAAC, Washington DC, USA, 2000.

[4] Ekbal, A. & Bandyopadhyay, S., *Lexicon Development and POS Tagging Using a Tagged Bengali News Corpus*, in Proceedings of the Twentieth International Florida Artificial Intelligence Research Society Conference (FLAIRS), pp. 261-262, 2007.

[5] Bojar, O., Diatka, V., Rychlý, P., Stranák, P., Suchomel, V., Tamchyna, A. & Zeman, D., *HindEnCorp – Hindi-English and Hindi-only corpus for machine translation*, Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC), pp. 3550-3555, 2014.

[6] Eichmann D., *The RBSE Spider – Balancing Effective Search Against Web Load*, Proceedings of the 1st International World Wide Web Conference, pp. 113-120, 1994.

[7] Heydon, A., & Najork, M., *Mercator: A Scalable*, *Extensible Web Crawler*, World Wide Web, **2**(4), pp. 219-229, 1999.

[8] Boldi, P., Codenotti, B., Santini, M., & Vigna, S., *Ubicrawler: A Scalable Fully Distributed Web Crawler*, Software: Practice and Experience, **34**(8), pp. 711-726, 2004.

[9] Blanvillain, O., Kasioumis, N. & Banos, V., *Blogforever Crawler: Techniques and Algorithms to Harvest Modern Weblogs*, in Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14), pp. 1-8, 2014.

[10] Casalnuovo, C., Suchak, Y., Ray, B. & Rubio-González, C., *Gitcproc: A Tool for Processing and Classifying Github Commits*, in Proceedings of the 26th ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA), pp. 396-399, 2017.

[11] Minhas, G. & Kumar, M., *LSI Based Relevance Computation for Topical Web Crawler*, Journal of Emerging Technologies in Web Intelligence, **5**(4), pp. 401-406, 2013.

[12] Wan, G., Ding, Y., Li, B. & Tan, X., *E&Vrobot: A Crawler of Education and Vocation*, Proceedings of the 9th International Conference on Computer Science and Education (ICCCSE), pp. 473-476, 2014.

[13] Bošnjak, M., Oliveira, E., Martins, J., Mendes Rodrigues, E. & Sarmento, L., *Twitterecho – A Distributed Focused Crawler to Support Open Research with Twitter Data*, Proceedings of the 21st Annual Conference on World Wide Web Companion, pp. 1233-1239, 2012.

[14] Del Vigna1, F., Cimino, A., Dell'Orletta, F., Petrocchi, M. & Tesconi, M., *Hate Me, Hate Me Not: Hate Speech Detection on Facebook*, Proceedings of the First Italian Conference on Cybersecurity (ITASEC17), pp. 86-95, 2017.

[15] Raad, B.T., Philipp, B., Patrick, H. & Christoph, M., *ASEDS: Towards Automatic Social Emotion Detection System Using Facebook Reactions*, Proceedings of IEEE 20th International Conference on High Performance Computing and Communications (HPCC), pp. 860-866, 2018.

[16] Guevara, E., *NoWaC: A Large Web-Based Corpus for Norwegian*, Proceedings of the NAACL HLT 2010 Sixth Web as Corpus Workshop, pp. 1-7, 2010.

[17] Jha, G.N., *The TDIL Program and the Indian Language Corpora Initiative (ILCI),* Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010), pp. 982-985, 2010.

[18] Kaur, J. & Saini, J.R., *PuPoCl : Development of Punjabi Poetry Classifier Using Linguistic Features and Weighting*, INFOCOMP: Journal of Computer Science, **16**(1-2), pp. 1-7, 2017.

[19] Jindal, S., Goyal, V. & Bhullar, J.S., *English to Punjabi Statistical Machine Translation using Moses (Corpus Based)*, Journal of Statistics and Management Systems, **21**(4), pp. 553-560, 2018.
[20] Rossum, G.V. & Drake, F.L., *Python 3 Reference Manual*, CreateSpace 2009.
[21] Urllib, https://docs.python.org/2/library/urllib.html (June 2021)
[22] BeautifulSoup, Accessed from https://www.crummy.com/software/BeautifulSoup/bs4/doc/(June 2021)