Data in Brief 55 (2024) 110751

Contents lists available at ScienceDirect

Data in Brief

journal homepage: www.elsevier.com/locate/dib

Data Article

In the heart of Swahili: An exploration of data collection methods and corpus curation for natural language processing

Bernard Masua, Noel Masasi*

College of Information and Communication Technologies (CoICT), University of Dar Es Salaam, Ali Hassan Mwinyi Road, Kijitonyama campus, Dar Es Salaam TZ 33335, Tanzania

ARTICLE INFO

Article history:
Received 2 March 2024
Revised 14 June 2024
Accepted 10 July 2024
Available online 17 July 2024

Dataset link: Swahili Corpus (Original data)

Keywords:
Text pre-processing
Swahili language
Corpus
Machine learning

ABSTRACT

Swahili corpus is a dataset generated by collecting written Kiswahili sentences from different sectors that deals with Kiswahili documents. Corpus of intended language is needed in Natural Language Processing (NLP) task to fit algorithm in order to understand that language before training the model. Swahili corpus dataset generated contained 1,693,228 sentences with 39,639,824 words and 871,452 unique words. Corpus exported in text file format with storage size of 168 MB. These sentences collected from different sources in different categories as follows: - Health (AFYA), Business and Industries (BIASHARA), Parliament (BUNGE), Religion (DINI), Education (ELIMU), News (HABARI), Agriculture (KILIMO), Social Media (MITANDAO), Non-Governmental Organizations (MASHIRIKA YA KIRAIA), Government (SERIKALI), Laws (SHERIA) and Politics (SIASA).

This abstract outlines the systematic data collection process employed for the creation of a Swahili corpus derived from multiple public websites and reports. The compilation of this corpus involves a meticulous and comprehensive approach to ensure the representation of diverse linguistic contexts and topics relevant to the Swahili language.

The data collection process commenced with the identification of suitable sources across various domains, including news articles, health publications, online forums, and Governmental public reports. Websites and platforms with pub-

* Corresponding author.
E-mail address: noeliasmasasi@gmail.com (N. Masasi).

https://doi.org/10.1016/j.dib.2024.110751

licly available Swahili content were systematically crawled and archived to capture a broad spectrum of linguistic expressions. Furthermore, special attention was given to reputable sources to maintain the authenticity of the corpus and linguistic richness. The inclusion of diverse sources ensures that the corpus reflects the linguistic nuances inherent in different contexts and registers within the Swahili language. Additionally, efforts were made to incorporate variations in domain dialects, acknowledging the linguistic diversity present in Swahili.

The potential for reusing this Swahili corpus is vast. Researchers, linguists, and language enthusiasts can leverage the diverse and extensive dataset for a multitude of applications, including NLP tasks such as sentiment analysis, textual data clustering, classifications tasks and machine translation. The Corpus can serve as training data for developing and evaluating NLP algorithms, including part-of-speech tagging, and named entity recognition. Also, text mining techniques can be applied to corpus and enable researchers to extract valuable insights, identify patterns, and discover knowledge from large textual datasets.

Specifications Table

Subject	Computer Science, Artificial intelligence
Specific subject area	Natural Language Processing, Swahili corpus dataset
Data format	Raw, Analyzed
Type of data	Text
Data collection	The scope of corpus was defined and relevant categories were identified by consulting subject matter experts and conducting literature reviews. Local websites and platforms known for providing authoritative information in Swahili language were reviewed and manually exploration of these sources was done to identify relevant documents and ensuring they align with the predetermined categories then download respective documents in PDF and DOCX file formats. A Python script was developed for automating the merging PDF and DOCX files, libraries such as PyPDF2 for PDFs and python-docx for DOCX files were used. A Python script was developed incorporating specific cleaning steps tailored to the characteristics of Swahili texts.
Data source location	Online contents from public institutions websites in Tanzania
Data accessibility	Repository name: **Mendeley Data** Data identification number: DOI: 10.17632/d4yhn5b9n6.2 Direct URL to data: https://data.mendeley.com/datasets/d4yhn5b9n6/2[1] Instructions for accessing these data: **None**

1. Value of the Data

- These datasets are important because they contribute to improving Swahili textual data preprocessing especially Swahili being a low resource language. For other languages such as English there are well documented resources for textual data pre-processing and can be accessed through different libraries which is not a case for Swahili [2].
- The datasets will benefit researchers, application developers and anyone interested in machine learning especially in NLP and works with Swahili textual data.
- These provided datasets can be used during data pre-processing stage [3] for NLP tasks such as Topic Analysis [4], classification [5], clustering [6] and Sentiment analysis [7] to provide the model with relative word patterns for context and similarity analysis.

- Also, these datasets can be updated and reused to fit into certain domain areas. The datasets can be modified for use in healthcare by adding medical terminology and patient interaction transcripts, or adapted for the legal field by including legal documents and court transcripts. Regular updates with domain-specific language data ensure the datasets remain relevant and useful for various applications such as natural language processing, machine learning, and linguistic research

2. Background

The original motivation behind compiling the Swahili NLP corpus stemmed from the recognition of a critical gap in the availability of comprehensive and diverse linguistic resources for the Swahili language. Swahili, being one of the official languages of several East African countries, plays a pivotal role in communication, education, and governance across the region [8]. Despite its significance, existing linguistic resources for Swahili were limited in scope and lacked the diversity required for robust NLP research [9].

The primary context for compiling this dataset was the need to address the challenges associated with the scarcity of high-quality Swahili language data for computational linguistic studies. The intention was to create a resource that not only captured the linguistic richness and variability of Swahili but also crossed multiple domains to accommodate a wide range of NLP applications.

In response to this motivation, a data collection process was designed, leveraging web crawling techniques to aggregate content from various public websites and reports. The goal was to create a corpus that reflected the reliable usage of Swahili across different contexts, including news, education, forums, and official reports. The compilation process aimed to ensure diversity, representativeness, and relevance to the current linguistic landscape of Swahili speakers.

3. Data Description

The repository contains several text files each corresponding to categories of Swahili textual contents (Table 1). The categories are Health (AFYA), Business and Industries (BIASHARA), Parliament (BUNGE), Religion (DINI), Education (ELIMU), News (HABARI), Agriculture (KILIMO), So-

Table 1
Dataset files description.

S/N.	Category	File Name	File Size (Bytes)	Number of Words	Number of Sentences	Number of Unique Words
1	Health	AFYA_Cleaned.txt	707,155	110,075	5775	9796
2	Business and Industries	BIASHARA_Cleaned.txt	433,124	78,229	2359	5831
3	Parliament	BUNGE_Cleaned.txt	3648,233	564,776	27,904	31,458
4	Religion	DINI_Cleaned.txt	5225,972	897,043	44,564	61,540
5	Education	ELIMU_Cleaned.txt	18,344,022	2904,474	156,123	156,238
6	News	HABARI_Cleaned.txt	32,401,254	5018,889	144,431	124,929
7	Agriculture	KILIMO_Cleaned.txt	4689,732	725,976	37,401	31,392
8	Social Media	MITANDAO_Cleaned.txt	173,720,366	27,187,801	1170,322	708,699
9	Non-Governmental Organizations	NGO_Cleaned.txt	891,304	137,151	7497	6944
10	Government	SERIKALI_Cleaned.txt	6239,181	962,341	49,382	37,347
11	Laws	SHERIA_Cleaned.txt	4579,059	698,048	34,768	29,359
12	Politics	SIASA_Cleaned.txt	2251,599	355,021	12,701	20,509
13	COMBINED	Swahili_Corpus_combined.txt	253,131,013	39,639,824	1693,228	871,452

cial Media (MITANDAO), Non-Governmental Organizations (MASHIRIKA YA KIRAIA), Government (SERIKALI), Laws (SHERIA) and Politics (SIASA).

4. Experimental Design, Materials and Methods

4.1. Identification of categories

The scope of corpus [10] was defined and above-mentioned relevant categories were identified by consulting subject matter experts from Institute of Kiswahili Studies. and conducting literature reviews on gathering data for corpus creation. These steps ensured a thorough and informed approach to gathering data for corpus creation.

4.2. Data collection from official sources

Local websites and platforms known for providing authoritative information in Swahili language were reviewed, whereby focus was mainly for governmental bodies, educational institutions, religious and other reputable organizations to ensure the authenticity of the content. In order to find relevant materials and ensure that they conform to the predefined categories, these sources have been systematically reviewed.

4.3. Download PDF and DOCX documents

After identifying relevant documents on official sources, the next step was to download respective documents that were in PDF and DOCX file formats. Where necessary a register describing the downloaded files, maintaining a record of file names, source URLs, was kept to facilitate traceability during subsequent stages of the corpus development.

The downloaded files were organised into category-specific folders to maintain a clear and structured dataset. This was done to ensure that the subsequent merging and cleaning processes can be executed separately and efficiently. Version control (using DVC) and backup mechanisms were deployed to prevent data loss and facilitate the tracking of changes made during corpus development.

4.4. Python script for merging documents

A Python script was developed for automating the merging PDF and DOCX files, libraries such as PyPDF2 [11] for PDFs and python-docx [12] for DOCX files were used to extract text content programmatically. The script was designed to iterate through category-specific folders, read each document, and concatenate the extracted text into a single text file for each category. The respective pseudocode of python script is shown on below pseudo-code.

Pseudo-code for merging files for each category

```
FUNCTION combine_texts_in_folder(folder_path)
    combined_text = “”
    FOR EACH file IN folder_path
        IF file HAS_EXTENSION “.pdf” THEN
            pdf_text = extract_text_from_pdf(file_path)
            combined_text = CONCATENATE(combined_text, pdf_text)
        ELSE IF file HAS_EXTENSION “.docx” THEN
            docx_text = extract_text_from_docx(file_path)
            combined_text = CONCATENATE(combined_text, docx_text)
        END IF
    END FOR
    RETURN combined_text
END FUNCTION
```

Merged text files were organised into folders named after the categories to preserve the inherent structure of the corpus.

4.5. Cleaning script

A Python script was developed incorporating specific cleaning steps tailored to the characteristics of Swahili texts. Cleaning steps included removing irrelevant whitespaces including tabs, remove multiple dots, remove roman numerals, converting to lowercases, remove URLs for website links, remove emails, remove punctuations, remove special characters and numbers. Most of the steps were accomplished using regular expressions capabilities provided by re python library to identifying and replace patterns that needs cleaning. The cleaning script was organized to process each category separately, ensuring targeted proper cleaning. The respective pseudocode of python script is shown on below pseudo-code.

Pseudo-code for corpus text cleaning

```
FUNCTION Clean_Corpus(list_text)
    new_list = []
    FOR EACH text IN list_text
        text = REMOVE_WHITESPACES(text)
        text = REMOVE_MULTIPLE_DOTS(text)
        text = REMOVE_ROMAN_NUMERALS(text)
        text = CONVERT_TO_LOWERCASE(text)
        text = REMOVE_URLS(text)
        text = REMOVE_EMAILS(text)
        text = REMOVE_PUNCTUATIONS(text)
        text = REMOVE_SPECIAL_CHATACTERS_NUMBERS(text)
        APPEND text TO new_list
    END FOR
    RETURN new_list
END FUNCTION
```

The main script iterated through the category folders, load the merged text data, and execute the cleaning script that finally save the cleaned text files back into their respective category folders. Error-handling mechanisms were incorporated within the script to manage unexpected situations, such as corrupted files or inconsistent formatting. The cleaned data was validated regularly to maintain textual data integrity.

4.6. Generate output category statistics

A Python script to extract descriptions and statistics from each cleaned text file was developed focusing on providing insights into the content of each category, offering valuable metadata for further analysis. The respective pseudocode of python script is shown on below pseudo-code.

Pseudo-code for generation corpus statistics

```
FUNCTION get_file_info(file_path, text)
    file_size = GET_FILE_SIZE(file_path)
    file_name = GET_FILE_NAME(file_path)
    words = SPLIT_TEXT_INTO_WORDS(text)
    sentences = SPLIT_TEXT_INTO_SENTENCES(text)
    num_words = COUNT_ELEMENTS_IN(words)
    num_sentences = COUNT_ELEMENTS_IN(sentences)
    num_unique_words = COUNT_UNIQUE_ELEMENTS_IN(words)
    RETURN file_size, file_name, num_words, num_sentences, num_unique_words
END FUNCTION
```

Metrics to be extracted were defined including; file Names, File Size, number of sentences, number of words, and number of unique words. sentiment analysis, or document length. Depending on the nature of the corpus topic keywords statistics were generated for each category. Fig. 1-12 shows word-cloud [13] visualization for most frequent words for each category.

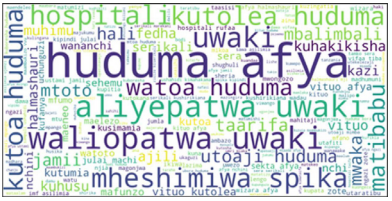

Fig. 1. Wordcloud for Health category.

Fig. 2. Wordcloud for Parliament category.

Fig. 3. Wordcloud for Religion category.

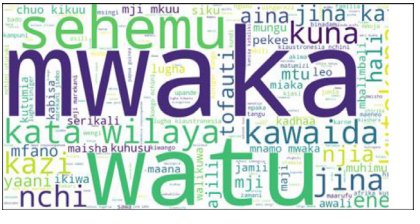

Fig. 4. Wordcloud for Education category.

Fig. 5. Wordcloud for Government category.

Fig. 6. Wordcloud for Politics category.

Fig. 7. Wordcloud for Laws category.

Fig. 8. Wordcloud for Agriculture category.

Fig. 9. Wordcloud for NGO category.

Fig. 10. Wordcloud for Social Media category.

Fig. 11. Wordcloud for News category.

Fig. 12. Wordcloud for Business and Industries category.

Finally, quality control measures were enforced throughout the corpus development process, including systematic checks on the downloaded documents, merged text files, and cleaned data by manually reviewing and ensuring that the data aligns with expectations and requirements. A feedback loop for continuous improvement was maintained by actively seeking input from domain experts, linguists, and other stakeholders.

Limitations

The corpus is limited to Swahili language textual data processing.

Ethics statement

The work does not involve human subject nor animals but ethical requirements for publication in Data in Brief journal are observed.

CRediT author statement

All authors contributed to the study conceptualization, design, material preparation, data collection, Software and analysis. **Bernard Masua**: Methodology, Validation, Writing - Review & Editing. **Noel Masasi**: Data curation, Writing- Original draft preparation, Visualization.

Data Availability

Swahili Corpus (Original data) (Mendeley Data).

Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] N. Masasi, B. Masua, Swahili Corpus, Mendeley Data V2 (2024), doi:10.17632/d4yhn5b9n6.2.
[2] B. Masua, N. Masasi, Enhancing text pre-processing for Swahili language: datasets for common Swahili stop-words, slangs and typos with equivalent proper words, Data Br. (2020), doi:10.1016/j.dib.2020.106517.
[3] B. Masua, N. Masasi, H. Maziku, B. Mbwilo, The impact of applying different pre-processing techniques on Swahili textual data using Doc2Vec, Nat. Lang. Process. Res. (2023), doi:10.55060/j.nlpre.230606.001.
[4] Y.A. Zelenkov, E.A. Anisichkina, Trends in data mining research: a two-decade review using topic analysis, Bus. Inform. (2021), doi:10.17323/2587-814X.2021.1.30.46.
[5] K. Kowsari, K.J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, D. Brown, Text classification algorithms: a survey, Information (2019), doi:10.3390/info10040150.
[6] M.H. Ahmed, S. Tiun, N. Omar, N.S. Sani, Short text clustering algorithms, application and challenges: a survey, Appl. Sci. (2023), doi:10.3390/app13010342.
[7] J. Cui, Z. Wang, S.B. Ho, E. Cambria, Survey on sentiment analysis: evolution of research methods and topics, Artif. Intell. Rev. (2023), doi:10.1007/s10462-022-10386-z.
[8] C.S. Shikali, Z. Sijie, L. Qihe, R. Mokhosi, Better word representation vectors using syllabic alphabet: a case study of Swahili, Appl. Sci. (2019), doi:10.3390/app9183648.
[9] D. Khurana, A. Koli, K. Khatter, S. Singh, Natural language processing: state of the art, current trends and challenges, Multimed. Tools Appl. (2023), doi:10.1007/s11042-022-13428-4.
[10] S. Crossley, A. Heintz, J.S. Choi, J. Batchelor, M. Karimi, A. Malatinszky, A large-scaled corpus for assessing text readability, Behav. Res. Methods (2023), doi:10.3758/s13428-022-01802-x.
[11] H. Seetha, V. Tiwari, K.R. Anugu, D.S. Makka, D.R. Karnati, A GUI based application for PDF processing tools using Python & CustomTkinter, Int. J. Res. Appl. Sci. Eng. Technol. (2023), doi:10.22214/ijraset.2023.48848.
[12] D. Murillo-Gonzalez and S. López, "Automation of the transformation process of publication formats in scientific journals through a Python script," 2023.
[13] I. Fellows, Wordcloud: Word Clouds, 2012 *R package version*.