PJEST

PJEST

Article

TPTS: Text pre-processing Techniques for Sindhi Language

Ali Nawaz[1, 2], Muhammad Nawaz[2], Noor Ahmed Shaikh[2], Samina Rajper[2], Junaid Baber[1], Muhammad Khalid[3]

[1]University of Balochistan, Quetta, Pakistan
[2] Shah Abdul Latif University (SALU) Khairpur, Pakistan
[3]HITEC University, Taxila, Pakistan
Correspondence: aliuob15@gmail.com

ABSTRACT: The Internet is a significant source of textual data, with users generating vast amounts of information through social media and news agencies daily. The extraction of meaningful information from large datasets is a challenging and costly process. Text pre-processing is a crucial initial step in any Natural Language Processing (NLP) task, as it can impact the overall performance of the study. The main objective of text pre-processing is to transform unstructured text into a linguistically meaningful (standard form) format, making extracting information for any text-processing task easier. This paper introduces TPTS, a model for text pre-processing in the Sindhi language. TPTS performs essential NLP tasks such as text tokenization, normalization, stop-word removal, stemming, and POS tagging for the Sindhi language. The Sindhi Text Corpus (STC), consisting of 1.5k Sindhi text documents collected from various online news websites, is used for experimentation. The TF-IDF approach is employed to identify high-frequency stop-words in the Sindhi language.

Furthermore, a rule-based system tags words with their part of speech in Sindhi input text. The ROUGE evaluation metric is used to assess the effectiveness of the proposed TPTS technique, achieving 89% accuracy on the STC corpus. The Sindhi language is spoken by over 30 million people globally, and the lack of adequate NLP tools and resources limits the development of technology and natural language applications that can benefit Sindhi speakers. The proposed TPTS model can aid in developing such applications, making it beneficial not only for text pre-processing tasks but also for other Sindhi language text-processing tasks such as text summarization, sentiment analysis, speech-processing applications, text mining, and information retrieval systems.

Keywords: Text pre-processing, natural language processing (NLP), ROUGE evaluation, TF-IDF weighting technique, rule-based model, Sindhi language.

Citation: Ali Nawaz, A. N., Muhammad Nawaz, Shaikh, N. A. . ., Rajper, S. . ., Baber, J. ., & Muhammad Khalid. TPTS: Text Pre-processing Techniques for Sindhi Language: Text Pre-processing Techniques. *Pakistan Journal of Emerging Science and Technologies (PJEST),* 4(3). https://doi.org/10.58619/pjest.v4i3.89

Academic Editor: Dr. M. Javaid Afzal

Received date: 18th Feb, 2023
Revised date: 27th May, 2023
Accepted date: 28th May, 2023
Published date: 28th June, 2023

1. Introduction:

Natural language processing (NLP) is the emerging field of artificial intelligence (AI), and linguistics [1]. The primary goal of the NLP is to provide computers with the ability to understand human (spoken and written) language [2]. Text pre-processing is critical in many NLP applications, including text summarization, classification, sentiment analysis, etc. The fundamental goal of text pre-processing is to transform the raw text into a format more amenable to analysis by downstream NLP models and algorithms [3]. Text pre-processing is challenging for Arabic script-based languages such as Sindhi due to a lack of linguistic resources, multi-token words, and the number of suffixes. Text pre-processing is critical in NLP applications for Sindhi and other Arabic script-based languages. However, several challenges, such as a need for linguistic resources, multi-token words, and complex morphological structures, make text pre-processing for these languages particularly challenging.

The Sindhi سنڌي Language is an Indo-Aryan language, with more than 75 million native speakers available worldwide [4]. Historically, the Sindhi language was used in obsolete

writing systems such as khudawadi, Khojki, Waranki, Gurumukhi, Landa, Devanagari, and Perso-Arabic. However, nowadays, only Perso-Arabic and Devanagari writing systems or scripts are widely used worldwide [4]. The Arabic script-based languages, like the Sindhi language, are written from right to left, with the text arranged horizontally [5, 6]. However, the numerals of the Sindhi language are written from left to right direction [7]. The Sindhi script has 52 alphabets, with 29 letters derived from Arabic, three letters derived from Persian, and 20 letters that are modifications or adaptations of the Arabic script to suit the phonetics of the Sindhi language better.

This study proposes TPTS: text pre-processing techniques for the Sindhi language text (s). This is the first-ever text pre-processing model for the Sindhi language. TPTS model aims to perform a series of pre-processing steps on Sindhi language text to clean and prepare it for further natural language processing tasks, such as text tokenization, text normalization, stop-word removal, stemming, and POS tagging.

The proposed TPTS model used an updated version of the extractive Sindhi corpus (ESC) named Sindhi text corpus (STC), which contains 1.5K text documents. The STC is generated from online Sindhi websites such as Online Indus News, Time News, and Awami Awaz. The TF-IDF approach is used to process the text in the Sindhi language, which generates high-frequency stop-words in the Sindhi language. Additionally, the model identified 183 prefix terms, 157 suffix terms, and 98 prefix-suffix terms as stem words of the Sindhi language.

The rule-based approach is used to tag words as part of speech (POS), essential for various natural language processing (NLP) tasks such as text summarization, sentiment analysis, speech-processing applications, text mining, and information retrieval systems. The proposed TPTS model achieved 89% accuracy based on the Sindhi text corpus (STC), demonstrating its effectiveness in processing Sindhi text. Moreover, this model can be used for any Sindhi text-processing task, making it a versatile tool for various NLP applications.

Challenges in Sindhi language text pre-processing

NLP and its subfield, such as text pre-processing for the Sindhi language, is one of the challenging tasks because Sindhi is a poor-resourced language having less computerized or digital resource availability online. However, significant advancements have been made in Arabic and Urdu compared to the Sindhi language. The Sindhi language has remained a poor-resourced language in computational linguistics and natural language processing [4].

The remaining parts of this paper are presented as follows: Section 2 includes the literature on Sindhi text pre-processing steps. Section 3 describes the research methodology and evaluation of the model (TPTS). Section 4 contains the experiments and results of the proposed model. Section 5 describes the conclusion and future studies of the research work.

2. Literature Review

The literature on text pre-processing in the Sindhi language has limited research study due to lake of NLP tools, open-source APIs, and publicly available online datasets.

I.N. Sodhar et al., 2021 [8] presented text tokenization for Sindhi language information retrieval. The dataset was crawled from the online Sindhi language Awami newspaper for text tokenization purposes, and the dataset is not available online for public use. The dataset consists of only one hundred forty (140) words (eight sentences) of Sindhi text.
J.A. Mahar et al., 2010 [9] proposed rule-based POS tagging for Sindhi text. The lexicon and word disambiguation rules are followed to develop Sindhi POS tagging. The dataset contains 26366 tagged words, and the corpus of tagged tokens was collected from different online Sindhi dictionaries. The results were evaluated using ROUGE metrics, and the Sindhi linguist verified the outcomes. The model achieved 96.28% accuracy from SPOS on the presented dataset.
W.A. Narego et al., 2016 [10] proposed a model for Sindhi word segmentation into morphemes. The proposed algorithm is designed to solve the segmentation problem of the Sindhi language. They have used the Sindhi corpus developed by Mahar et al., 2014 [11], which contains 105733 words. The algorithm tested 109 compound terms, 179 prefix terms, 1343 suffix terms, and 50 prefix-suffix terms in the Sindhi language. The overall segmentation error rate was calculated 5.02%.
M.R. Shah et al., 2016 [12] presented a Sindhi stemmer for IR systems by applying a rule-based stripping approach. The proposed rule-based algorithm depends on the developed lexicon and linguistic rules. The Sindhi corpus Mahar et al., 2014 [11], is used for experiments, containing 86,733 words. The 5327 words have been considered for the lexicon, where 2142 words were found to have prefixes and suffix morphemes of the Sindhi language. The proposed Sindhi stemmer performed with 84.85% accuracy on the developed dataset.
M. Osman Hegazi et al., 2021 [13] presented text pre-processing for Arabic text on social media. The proposed model process the data in four steps: data collection, cleaning, enrichment, and availability. The preprocessed text is stored in the corpus, where the meaningful data can be extracted by applying an algorithm, and it becomes easier to query and analyze the data, allowing for more efficient and effective processing. The processed Arabic text can also be valuable for topic classification and sentiment analysis.
Anandarajan et al., 2019 [14] presented a chapter on text pre-processing. The primary purpose of text pre-processing is to cleanse text data by applying the following steps: tokenizing, standardizing, cleaning, removing stop words, and stemming. Further advanced techniques such as n-grams, part-of-speech tagging, and custom dictionaries are applied for efficient results.
Anoual EI Kah et al., 2021 [15] proposed the effects of text pre-processing on Arabic text classification. Several text classification models are applied for better results of text pre-processing. They have produced 50,000 articles dataset of different categories: culture, politics, sport, economy, health, and technology. The following steps: stop Words (SWs) elimination, stemming, and lemmatization were performed for cleaning the text. Further, TF-IDF and Chi-square were used for feature selection and feature extraction. Finally, the following algorithms (SVM, DT J48 and NB) were implemented for text classification.

3. Research Methodology

The proposed TPTS: Text pre-processing techniques for the Sindhi language is shown in Fig. 1. The model comprises three steps: in the initial stage, the dataset is developed and named as Sindhi text corpus (STC). Secondly, text pre-processing is applied to input

Sindhi text, and in the third step, the model generates clean text, which is ready for further text processing task (s).

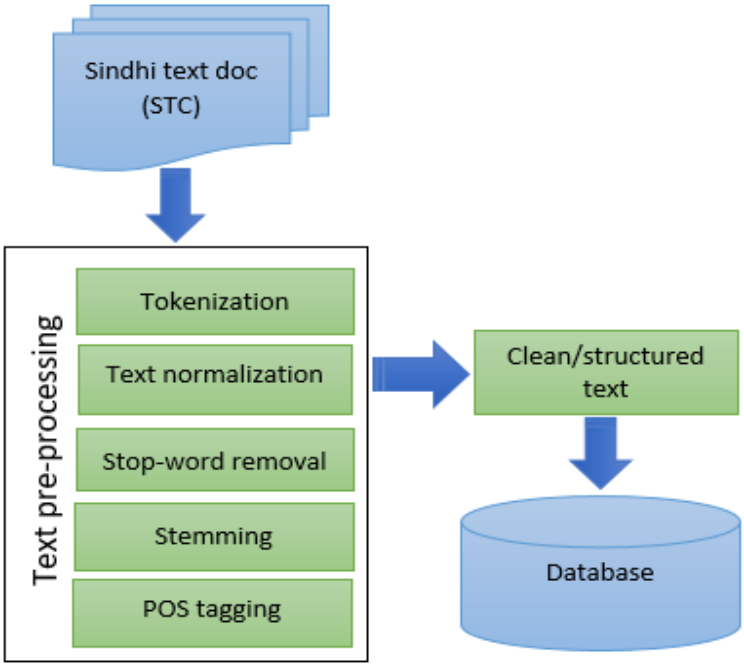

Fig. 1: The proposed framework of the model

3.1 Development of dataset

The corpus for any research work has inevitably essential for experiments and evaluation. In this research, the online publicly available dataset is used named Extractive Sindhi Corpus (ESC), developed by A. Nawaz et al., 2022 [16]. The ESC contains 1.2K articles on different topics such as; politics, sports, health, tourism, and national and international news documents crawled from various online Sindhi websites such as; Online Indus News, Time News, and Awami Awaz. The ESC was developed and evaluated for Sindhi text summary generation. In this research, the ESC is assessed and used for the following purposes: text tokenization, text normalization, stop-word removal, stemming, and POS tagging. The ESC contains a 1.2K document dataset. In this research, we have extended the ESC corpus and added 300 more text documents for more accurate results generation. The proposed corpus, named as Sindhi text corpus (STC), contains 1.5K text documents. The STC includes 1.5K documents, 43,500 sentences, 670,505 tokens, and 36,000 unique words. Table 4 shows the statistics of the STC dataset.

3.2 Text pre-processing

Text pre-processing is the initial step toward any text processing and NLP task [3]. Text pre-processing includes: 1. Text tokenization, 2. Text normalization (special characters, HTML tags, punctuation marks, numbers, and non-Sindhi words removal), 3. Stop-words removal 4. Stemming, and 5. POS tagging:

3.2.1 Text tokenization

Text tokenization is the initial step toward text pre-processing[16, 17]. The following two operations are required to convert the text into tokens: convert the paragraph into sentences and split the sentences into tickets [18]. In this research TPTS model automatically converts the section into sentences by using the following delimiters: full stop (-), question mark (?), and exclamation mark (!), whereas white-spaces ( ), commas (,), and semicolons (;) are used to split sentences into tokens.

3.2.2 Text normalization

Text normalization is the process of removing: memorable characters, HTML tags, punctuation marks, numbers, and non-Sindhi words, from input documents. The normalization process improves the performance of any NLP task, such as text-to-speech synthesis, speech recognition, information extraction, text summarization, sentiment analysis, and machine translation [19]. The list of the following symbols: (special characters, HTML tags, punctuation marks, numbers, and non-Sindhi words) is added to the proposed TPTS model, eliminating these symbols/characters from the input Sindhi document during the processing step.

3.2.3 Stop-words removal

The stop-words are less meaningful and most frequent words in any document in any language, such as Arabic [20], Persian [21], Urdu [22], and Sindhi [23]. The stop-words have the least semantic values in the text. The stop-words have an insignificant effect on the reader, whereas the stop-words help to format and complete the sentence. The stop-words mainly include pronouns, conjunctions, and prepositions [24].
In this research, the term frequency-Inverse document frequency (TF-IDF) is used to calculate the weight of stop-words in the document. The STC corpus contains 670,505 words; among these words, 522 high-frequency words are identified through the TF-IDF technique as stop-words of the Sindhi language. Table I shows the list of some high-frequency stop-words in the Sindhi language.

Table I: An example of high-frequency stop-words extracted from STC using TF-IDF.

Stop-words (SW)	Frequency	Stop-words (SW)	Frequency
جي	3845	آهي	3721
آهن	3578	جو	3111
آهيون	2901	سان	2709
کان	2665	تو	2423
اهو	2103	ڪري	1953
کي	1875	ڪئي	1133

The TF-IDF technique is used in different fields of study, such as information retrieval, machine learning, and text mining [16]. The TF-IDF is a statistical approach mostly used to calculate the weight of any word in the document (s) [23].

Term frequency (TF): $tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$

$n_{i,j}$: The number of occurrences of the term in document d_j

$\sum_k n_{k,j}$: Number of occurrences of all terms in document d_j

Inverse Document Frequency (IDF):$idf_i = log \frac{|_D|}{|\{d_j : t_j \in |d_j\}|}$

$|$D $|$: total document in a data set (SSC)

$|\{d_j : t_j \in d_j\}|$: the number of documents where t_i appears.

Term Frequency-Inverse Document Frequency (TF-IDF): tf- $idf_{i,j} = tf_{i,j} * idf_i$

3.2.4 Stemming

Stemming converts the inflected words to their root or stem form [24, 25]. A variety of checking approaches is available in different languages, such as English [26], Arabic[27, 28], Persian [29], and Urdu [30]. Also, several stemming approaches are available for the Sindhi language [12, 25, 31, 32]. The A. A. Sattar et al., 2021 [25] proposed Sindhi stemmer and identified 72 prefix terms, 150 suffix terms, and 41 prefix-suffix terms; these terms were used in the proposed research. Further, some more inflected words are identified, such as 183 prefix terms, 157 suffix terms, and 98 prefix-suffix terms, which are acquired from the STC dataset. Table II shows some inflected words followed by their root or stem form.

Table II: Sindhi-inflected text followed by its stem form.

Word	Prefix	Stem	Suffix
سڏواخلاق	سڏو	اخلاق	--
خوشگوار	--	خوش	گوار
بدقسمت	بد	قسمت	--
سدائين	سدا	ئين	--
آرامده	--	آرام	د ه

3.2.5 Part of speech tagging

Part of speech (POS) tagging is a process of sequential labeling of each word with the respective label [33]. The process to assign syntactic categories to each word as noun, pronoun, verb, and adjective [9, 34]. POS tagging for resource-poor language Sindhi is one of the challenging tasks due to the Sindhi language being morphologically vibrant. In this research, the rule-based part of speech tagging is used to tag the words of the input document. J.A. Mahar et al., 2010 [9] proposed a Sindhi POS tagging corpus containing 26366 tagged comments.

This research uses the rule-based approach to tag each word of the input Sindhi document with the respective tag. The proposed rule-based POS tagging is performed 90% accuracy, while the experiments are performed on Sindhi text corpus (STC). The proposed model rule-based POS tagging results are compared with the already tagged dataset [9]. The STC dataset tokens are classified based on eight parts of speech: noun, pronoun, verb, adverb, adjective, conjunction, preposition, and interjection. Table III shows Sindhi words with respective tags and descriptions, followed by English translations.

Table III: POS tagging of Sindhi words with English translation.

Sindhi Word	English translation	POS tagging	Description
ڊاڪٽر	Doctor	NN	Noun, singular or mass
کيڈان	Play	VBD	Verb, past tense
خوبصورت	Beautiful	JJ	Adjective
هي	He	PRP	Personal pronoun
سنڌي	Sindhi	NNP	Noun, pronoun, singular

In this step, the TPTS model generates clean text for further processing. The various field of research requires text pre-processing while performing operations such as information retrieval system, text summarization, text mining, sentiment analysis, and text processing [20, 35]. The abstract flow of pre-processing steps is shown in Fig. 2.

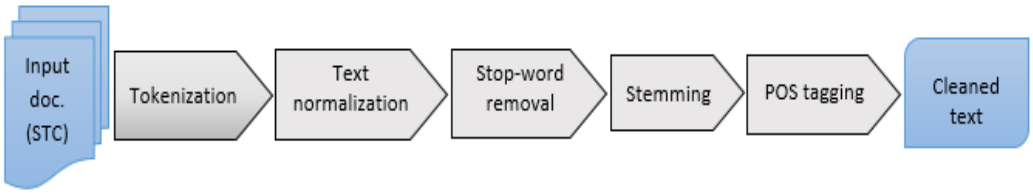

Fig. 2: Text pre-processing steps

4. Experiments and results

The evaluation process for this research presented several challenges due to limited online resources available for text pre-processing in the Sindhi language. Publicly available APIs, datasets, and stop-word lists were scarce. Therefore, some aspects of the research were performed manually by experts in the Sindhi language. The Sindhi Text Corpus (STC), which contains 1.5K text documents in Sindhi, was used for evaluation and experimentation. The STC corpus consists of 670,505 total tokens and 36,000 unique words. Table IV presents the statistics of the STC dataset.

Table IV: Sindhi text corpus dataset statistics, STC contains 1.5K text documents of the Sindhi language.

Corpus	Total documents	Total sentences	Total words	Total Unique words
STC	1500	43,500	6,70,505	36,000

The study utilized the Python library "Urduhack" to tokenize documents in Python 3.7. To determine the significance of stop-words, the team applied the TF-IDF technique, and a rule-based approach was employed to tag the words with their corresponding parts of speech. The proposed TPTS model was evaluated on the STC dataset using the ROUGE evaluation model and achieved an impressive overall performance (F-score) of 89%. Table V presents the extended version of the TPTS model, considering precision, recall, and f-score. Notably, the team's approach showed a significant improvement compared to the baseline performance of 81%, indicating the potential of the TPTS model in accurately analyzing Urdu text. The baseline model is the classical approach without text pre-processing. The classical approach directly processes the raw data for the desired task. The dataset, list of stop-words, and scripts are provided on GitHub: (https://github.com/AliNawazUoB/sindhi-TPTS) for peer researchers.

Table V: Using the ROUGE evaluation measure, the model's overall performance on Sindhi text corpus (STC).

Approaches	Precision	Recall	F-score
TPTS	86%	93%	89%
Baseline	75%	89%	81%

Fig. 3 shows the results of the ROUGE evaluation in terms of precision, recall, and F-score, where the blue line indicates accuracy, the red line shows the memory, and the green line indicates the F-score of the model. The overall performance of the TPTS model measured 89% performance (f-score) on the STC dataset.

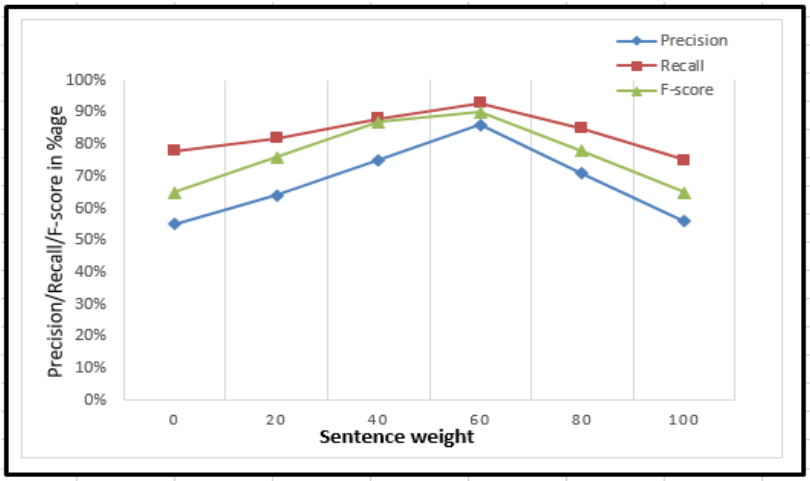

Fig. 3: shows the model's overall performance in precision, recall, and f-score.

5. Conclusion

This study presented the Text Pre-Processing Techniques for Sindhi (TPTS) model, which performs several text pre-processing steps, including tokenization, normalization, stop-word removal, stemming, and POS tagging on input Sindhi text files. The team generated the Sindhi text corpus (STC) by collecting 1.5K text documents from various online Sindhi news websites. Implementing the TF-IDF approach led to the identification of 522 high-frequency Sindhi stop-words, the highest number to date in the Sindhi language. Furthermore, the team utilized the Sindhi stemming text proposed by A. A. Sattar et al. (2021) and identified 183 prefix terms, 157 suffix terms, and 98 prefix-suffix terms for text normalization [25]. The rule-based approach was utilized for POS tagging. The TPTS model showed significant improvement over the baseline method, achieving an overall performance of 89% compared to 81%. Overall, the study contributes to the development of compelling text pre-processing techniques for Sindhi language analysis, providing valuable insights for future research in this field.

Author statement: Ali Nawaz: the principal author of this research work, has contributed significant efforts to accomplish this research. Further, Muhammad Nawaz Joyo (Ph.D. enrolled scholar), Noor Ahmed Shaikh (Ph.D.), Samina Rajper (Ph.D.), Junaid Baber (Ph.D.), and Muhammad Khalid (Ph.D. enrolled scholar) are the coauthors of the research and provided technical support for finalizing the research.

Funding: The authors received no specific funding or financial support for this research.

Conflict of interest: The authors declare no conflict of interest.

REFERENCES

[1] A. Reshamwala, D. Mishra, and P. Pawar, "Review on natural language processing," *IRACST Engineering Science and Technology: An International Journal (ESTIJ)*, vol. 3, pp. 113-116, 2013.

[2] T. P. Nagarhalli, V. Vaze, and N. Rana, "Impact of machine learning in natural language processing: A review," in *2021 third international conference on intelligent communication technologies and virtual mobile networks (ICICV)*, 2021, pp. 1529-1534.

[3] M. Shamsfard, S. Kiani, and Y. Shahedi, "STeP-1: standard text preparation for Persian language," in *Proceedings of the Third Workshop on Computational Approaches to Arabic-Script-based Languages (CAASL3)*, 2009.

[4] R. Motlani, "Developing language technology tools and resources for a resource-poor language: Sindhi," in *Proceedings of the NAACL Student Research Workshop*, 2016, pp. 51-58.

[5] M. A. Dootio and A. I. Wagan, "Syntactic parsing and supervised analysis of Sindhi text," *Journal of King Saud University-Computer and Information Sciences*, vol. 31, pp. 105-112, 2019.

[6] N. A. Shaikh, G. A. Mallah, and Z. A. Shaikh, "Character segmentation of Sindhi, an Arabic style scripting language, using height profile vector," *Australian Journal of Basic and Applied Sciences,* vol. 3, pp. 4160-4169, 2009.

[7] Y. A. Solangi, Z. A. Solangi, A. Raza, N. A. Shaikh, G. A. Mallah, and A. Shah, "Offline-printed sindhi optical text recognition: Survey," in *2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, 2018, pp. 1-5.

[8] I. N. Sodhar, J. Hussain, A. Buller, and A. Sodhar, "TOKENIZATION OF SINDHI TEXT ON INFORMATION RETRIEVAL TOOL," *PAKISTAN J. Emerg. Sci. Technol,* vol. 1, pp. 10-16, 2020.

[9] J. A. Mahar and G. Q. Memon, "Rule based part of speech tagging of sindhi language," in *2010 International Conference on Signal Acquisition and Processing*, 2010, pp. 101-106.

[10] W. A. Narejo, J. A. Mahar, S. A. Mahar, F. A. Surahio, and A. K. Jumani, "Sindhi morphological analysis: an algorithm for sindhi word segmentation into morphemes," *Int. J. Comput. Sci. Inf. Secur,* vol. 293, 2016.

[11] S. Mahar, "Comparative Analysis of Vowel Restoration for Arabic Script Based Languages Using N-Gram Models," MS Thesis, Department of Computer Science, Shah Abdul Latif University …, 2014.

[12] M. SHAH, H. Shaikh, J. MAHAR, and S. MAHAR, "Sindhi stemmer for information retrieval system using rule-based stripping approach," *Sindh University Research Journal-SURJ (Science Series),* vol. 48, 2016.

[13] M. O. Hegazi, Y. Al-Dossari, A. Al-Yahy, A. Al-Sumari, and A. Hilal, "Preprocessing Arabic text on social media," *Heliyon,* vol. 7, p. e06191, 2021.

[14] M. Anandarajan, C. Hill, T. Nolan, M. Anandarajan, C. Hill, and T. Nolan, "Text preprocessing," *Practical text analytics: Maximizing the value of text data,* pp. 45-59, 2019.

[15] A. El Kah and I. Zeroual, "The effects of pre-processing techniques on Arabic text classification," *Int. J,* vol. 10, pp. 1-12, 2021.

[16] A. Nawaz, R. A. Shaikh, R. H. Arain, S. Rajper, J. Baber, and M. M. Baidani, "Text Summarizer for Sindhi Language," *Available at SSRN 4288269*.

[17] S. Mohtaj, B. Roshanfekr, A. Zafarian, and H. Asghari, "Parsivar: A language processing toolkit for Persian," in *Proceedings of the eleventh international conference on language resources and evaluation (lrec 2018)*, 2018.

[18] A. Nawaz, M. Bakhtyar, J. Baber, I. Ullah, W. Noor, and A. Basit, "Extractive text summarization models for Urdu language," *Information Processing & Management,* vol. 57, p. 102383, 2020.

[19] C. Zhang, T. Baldwin, H. Ho, B. Kimelfeld, and Y. Li, "Adaptive parser-centric text normalization," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2013, pp. 1159-1168.

[20] A. Qaroush, I. A. Farha, W. Ghanem, M. Washaha, and E. Maali, "An efficient single document Arabic text summarization using a combination of statistical and semantic features," *Journal of King Saud University-Computer and Information Sciences,* vol. 33, pp. 677-692, 2021.

[21] M. Sadeghi and J. Vegas, "Automatic identification of light stop words for Persian information retrieval systems," *Journal of Information Science,* vol. 40, pp. 476-487, 2014.

[22] A. Daud, W. Khan, and D. Che, "Urdu language processing: a survey," *Artificial Intelligence Review,* vol. 47, pp. 279-311, 2017.

[23] M. A. Dootio and A. I. Wagan, "Development of Sindhi text corpus," *Journal of King Saud University-Computer and Information Sciences,* vol. 33, pp. 468-475, 2021.

[24] R. Al-Shalabi, G. Kanaan, J. M. Jaam, A. Hasnah, and E. Hilat, "Stop-word removal algorithm for Arabic language," in *Proceedings. 2004 International Conference on Information and Communication Technologies: From Theory to Applications, 2004.*, 2004, p. 545.

[25] A. A. Sattar, S. Abbasi, M. U. Rahman, A. Baig, and M. Nizamani, "Sindhi stemmer using affix removal method," *International Journal,* vol. 10, 2021.

[26] P. Willett, "The Porter stemming algorithm: then and now," *Program,* vol. 40, pp. 219-223, 2006.

[27] A. Al-Omari, B. Abuata, and M. Al-Kabi, "Building and benchmarking new heavy/light Arabic stemmer," in *The 4th International conference on Information and Communication systems (ICICS'13)*, 2013, pp. 17-22.

[28] S. Khan, W. Anwar, U. Bajwa, and X. Wang, "Template based affix stemmer for a morphologically rich language," *International Arab Journal of Information Technology (IAJIT),* vol. 12, 2015.

[29] J. Mehrad and S. Berenjian, "Providing a Persian language singular-stemmer system (RICeST Stemmer)," 2011.

[30] R. Kansal, V. Goyal, and G. S. Lehal, "Rule based urdu stemmer," in *Proceedings of COLING 2012: Demonstration Papers*, 2012, pp. 267-276.

[31] S. D. Makhija, "A study of different stemmer for sindhi language based on devanagari script," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2016, pp. 2326-2329.

[32] B. Nathani, N. Joshi, and G. Purohit, "Design and development of unsupervised Stemmer for Sindhi language," *Procedia Computer Science*, vol. 167, pp. 1920-1927, 2020.

[33] W. Ali, R. Kumar, Y. Dai, J. Kumar, and S. Tumrani, "Neural Joint Model for Part-of-Speech Tagging and Entity Extraction," in *2021 13th International Conference on Machine Learning and Computing*, 2021, pp. 239-245.

[34] I. N. Sodhar, A. H. Jalbani, M. I. Channa, and D. N. Hakro, "Parts of speech tagging of Romanized Sindhi text by applying rule based model," *IJCSNS*, vol. 19, p. 91, 2019.

[35] S. Vijayarani, M. J. Ilamathi, and M. Nithya, "Preprocessing techniques for text mining-an overview," *International Journal of Computer Science & Communication Networks*, vol. 5, pp. 7-16, 2015.