Indonesian Journal of Electrical Engineering and Computer Science
Vol. 25, No. 3, March 2022, pp. 1501~1507
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v25.i3.pp1501-1507 ❐ 1501

Automatic construction of generic stop words list for hausa text

Abdulkadir Abubakar Bichi, Ruhaidah Samsudin, Rohayanti Hassan
School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia (UTM), Johor, Malaysia

Article Info

Article history:

Received Aug 28, 2021
Revised Dec 27, 2021
Accepted Jan 21, 2022

Keywords:

Document pre-processing
Hausa stop words
Natural language processing
Stop words
Text mining

ABSTRACT

Stop-words are words having the highest frequencies in a document without any significant information. They are characterized by having common relations within a cluster. They are the noise of the text that are evenly distributed over a document. Removal of stop words improve the performance and accuracy of information retrieval algorithms and machine learning at large. It saves the storage space by reducing the vector space dimension, and helps in effective documents indexing. This research generated a list of hausa stop words automatically using aggregated method by combining frequency and statistics methods. The experiments are conducted using a primarily collected hausa corpus consisting of 841 hausa news articles of size 646862 words and finally a list of distinct 81 hausa stop words is generated.

Corresponding Author:

Ruhaidah Samsudin
School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia (UTM)
Johor, Malaysia
Email: ruhaidah@utm.my

1. INTRODUCTION

Stop-words are words with the highest frequencies in a document without any significant information [1]. They are characterized by having common relations within a cluster [2]. The presence of stop words has insignificant effect on the overall semantic of sentences, usually used to satisfied the grammatical rule of the language [3]. They were described as noise which is evenly distributed over a document [4]. Removal of stop-words abridge the total bytes of the documents, therefore speedup the processing time of most information retrieval (IR) applications such as automatic text summarization, questions answering and recommendation system. It is described as a way of improving the performance of information retrieval in general [5]-[7] and such removal better the performance of some applications like search engines [8], text classification [9], detection of keyphrases [10], automatic detection of grammatical errors [11], computation of semantic similarity [12], identification sequence patterns [13], spam detection in e-mail [14], detection and removing unwanted videos [15], detection of hate speech [16], identification of named entity [17]. Non removal of stop words affects the process of automatic selecting keywords or important phrases from a document [18], [19]. It is the most vital preprocessing activity in Information Retrieval and Artificial Intelligence researches [20], [21].

The stop-words are categorized into two: the grammar-specific stop-words and the domain-specific stop-words. The grammar-specific stop-words includes the list of language pronouns, prepositions, conjunctions, adjectives, adverbs and prefixes [20]. The domain-specific stop-words is specifically to a particular domain of information. Stop-words are generated using diverse approaches including, dictionary, machine learning, words frequency, entropy-based, statistical-based and part of speech (PoS) approaches. The simplest approach for creating stop-words is frequency method [5]. The method generates stop words by computing the frequency of words, those with the highest frequencies in the corpus are consider the stop words. The statistical method determines the average probability and variance of words, the words with the highest probabilities and lowest variances are consider the stop words. The information theory model works by considering the information weight of a word. The method generates the stop words by considering the entropy of words, those with lowest entropy are made of the list.

This research generated a list of stop words for hausa language. Hausa is a Chadic language, widely spoken in West Africa by about one hundred and fifty million (150,00,000) peoples at either first or second language. The language is the most widely spoken indigenous language in West Africa. Its native speakers spread across southern Niger, northern Nigeria, Ghana and Northern Cameroun. It is also used for trades in other places like Equatorial Africa, Chard and Sudan. The hausa is the largest ethnic group in west and north-central Africa. The significant number of hausa speakers are also found in Saudi Kingdom, Benin republic, ivory coast and togo. Many international media including british broadcasting corporation (BBC), voice of america (VOA), radio france international (RFI), and china radio international (CRI) broadcast ranges of programs in the language. There are a lot of literature related to religion and traditions written in the languages, which may be of interest to many readers across the globe. The remaining parts of the paper are organized as follows. Section 2 presents the review of the related works. Section 3 presents the methodology of the research. Section 4 presents the results of the experiments. Section 5 presents the research conclusion and future research directions.

A comprehensive list of stop-words has been developed for English language longtime ago. Recently, various researches proposed a stop-words list for other languages such as Hindi [3], Malay [22], Arabic language [21], [23], [24], Thai [25], Gujarati language [26], Urdu text [27]. Similarly, Girmaw and Khedkar [20] generated a stop-words list for Amharic language using aggregated-based technique, by combining word frequency and entropy method. In the paper, Raulji and Saini [28] generated stop-words list for Sanskrit language using hybrid method, the method used automatic algorithm with some involvement of human experts. In the paper, Asubiaro [29] generated stop-words list for Yoruba language using entropy-based approach. Similarly, another list was generated for Yoruba language using aggregated method by combining frequency and words entropy techniques [4].

Stop-words list was automatically generated for Egyptian dialect using frequency method [30]. The aggregate method was used for generation of stop-words list for Persian language by combining statistical and similarity function approaches [31]. A deterministic finite automaton was used for generation of stop-words for Hindi text [32]. More so, machine learning algorithm was used for automatic generation of Bengali stop words [33]. Similarly, Sadeghi and Vegas [18] automatically generated a list of light stop-words for Persian text using aggregated approach by combining frequency, statistics and entropy methods. Some researches focus on domain specific stop words, Na and Xu [5] created a stop-words list for Chinese patents using both frequency and statistical approaches. The stop words list was also generated for technical language for the use of engineering and related field of knowledges [34].

2. METHOD

The details of research methods and the description of dataset used in the research are presented in the following subsections. The research used frequency and statistics approach to generates the hausa stop words. The dataset is primarily collected from varous hausa news websites, the final corpus is comprised of 841 hausa text news articles.

2.1. Dataset

A hausa corpus was primarily collected for the experiments. The task of corpus creation nowadays is a challenging task due to the people style of writing. Many people write text on internet using non-standard styles including too many abbreviations and mixing languages. To minimize that, the corpus was only consider from standard hausa news websites. The corpus is comprised of 841 hausa news articles from BCC hausa; VOA hausa; RFI hausa; aminiyya newspaper and hausa leadership newspaper. The text file was converted to UTF-8 format for Python compatibility, as illustrated in Table 1.

Table 1. Description of dataset

Corpora	Number of documents	Total words
Corpus 1	130	40869
Corpus 2	323	105436
Corpus 3	477	146601
Corpus 4	602	166813
Corpus 5	841	187143

2.2. Frequency method

The frequency method was used for creation of hausa stop-words in the research. The term frequency of a word is simply referred to the word count or number of its occurrence in a given corpus. Mathematically the term frequency of a word is determined as (1):

$$\mathrm{tf} = (\mathrm{tf}, \mathrm{c}) / (\Sigma \mathrm{ft}, \mathrm{c}) \tag{1}$$

where, $\boldsymbol{tf}$, c is Term frequency in a corpus and $\Sigma\ \boldsymbol{ft}$, c total word number of terms of corpus. The stop-words are generated using a frequency method as follows:

- Perform sentence and word level tokenization
- Generate words frequencies in the corpus
- Sort words based on their frequencies
- Select the top rank words

2.3. Statistics

The hausa stop-words are further generated in the research using statistical method in the following steps:

- Perform sentence and word level tokenization
- Calculate each word's SAT value in the corpus
- Sort the word according to their SAT in descending order
- Extract the words with high SAT as candidates, and filter them manually

Suppose the corpus D={di}, 1=<i<=N. N refers to the count of document. The set of words in corpus is denoted as W={wj}. The average probability MP of word wj in D is:

$$MP(Wj) = \frac{\sum_{1 \le i \le N} Pij}{N} \tag{2}$$

pij is the frequency probability of wj in di. In other words, pij equals to wj 's frequency in di divided by the number of words in di. If a word has a high MP value, it implies that this word occurs frequently in the whole corpus. The variance VP of wj in D is:

$$VP(Wj = \frac{\sum_{1 \le i \le N} Pij - MP(Wj)2}{N}) \tag{3}$$

If a word has a low VP value, it implies that this word occurs uniformly in the whole corpus. The SAT of wj in D is:

$$SAT(Wj) = \frac{MP(Wj)}{\sqrt{VP(Wj)}} \tag{4}$$

If a word has a high SAT value, it implies that this word occurs frequently and uniformly in the whole corpus. The word like this is very likely to be a stop word. The intersection of words appeared in both lists using frequency-based and statistics method is taken and consider as the final stop words, as illustrated in Figure 1.

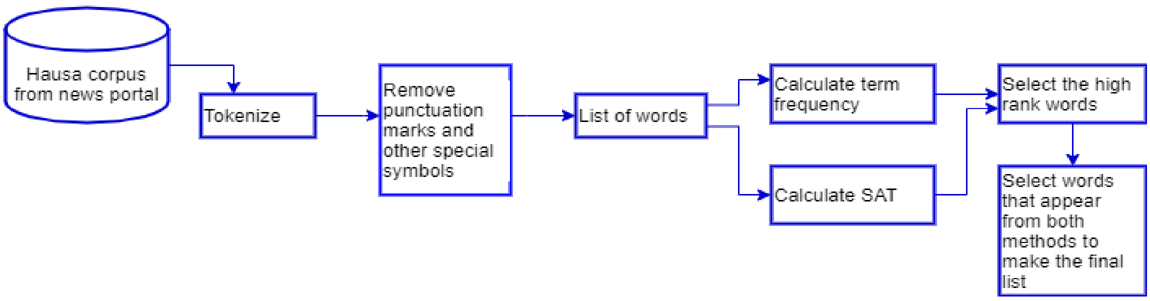

Figure 1. Architecture of the proposed work

3. RESULTS AND DISCUSSION

Five different sizes of corpora were used for the experiments. The corpus with 40869 total words size; 105436 total words size; 146601 total words size; 166813 total words size; and 187143 total words size. The experiments produced five different stop words lists, as illustrated in Table 2.

The experiments were conducted on the same dataset using the statistical method and the results are presented in Table 3. The results presented the variance of individual term using different corpus sizes. The list is madeup 100 hausa words with their variance.

Table 2. Top 20 highest frequency words using frequency method under different corpora

40869 words	105436 words	146601 words	166813 words	187143 words
Term: Count	Term: Count	Term: Count	Term: Count	Term: Count
da:3258	da:8431	da: 11164	da:12835	da: 14427
ya:1177	ya:2826	ta: 4140	ta:4740	ya: 5076
ta:1028	ta:2757	ya: 4047	ya:4501	ta: 4987
na:686	na:1848	na: 2514	na:2894	na: 3324
ba:502	ba: 1296	ba: 1701	ba:1807	ba: 1971
yi:457	yi:1161	kuma: 1595	kuma:1757	kuma: 1848
su:421	kuma:1112	yi: 1426	yi:1526	yi: 1828
ce:421	ne:1000	ne: 1311	ne:1431	ne: 1508
kuma:418	ce:908	ce: 1101	su:1193	ce: 1330
ne:397	su:879	su: 1078	ce:1189	su: 1329
za:317	ke: 743	za: 922	ke:1035	ke: 1281
ke:286	za:711	ke: 914	daga:1032	suka: 1144
daga:282	daga:647	daga: 875	suka:1008	daga: 1076
suka:239	shi:634	dan: 861	za:986	mai: 1043
shi:231	mai:626	mai: 833	mai:918	za: 1006
wa:228	suka: 614	cewa: 824	dan:910	cewa: 987
sun:222	kan: 593	suka: 798	cewa:890	yan: 974
ga:220	sun: 591	shi: 796	shi:845	sun: 957
cikin:217	wa:586	aka:738	sun: 841	kan: 932
yan:207	aka:576	kan:725	kan: 840	aka: 931

Table 3. Top 20 words with the highest spread/distribution under different corpora

187143 words	166813 words	146601 words	105436 words	40869 words
Term: Variance	Term: Variance	Term: Variance	Term: Variance	Term: Variance
da:29886.18	da:25886.18	da:23005.28	da:22746.10	da:20640.80
ya:24776.76	ta:21176.76	ya:20036.76	ya:19903.81	ya:17005.60
ta:23910.11	ya:20995.11	ta:18995.71	ta:16500.20	ta:14005.20
kuma:14633.61	na:13611.00	na:13004.21	na:12401.05	na:10650.10
ba:9062.23	yi:8057.23	ba:7800.30	ba:7091.10	ba:6005.11
yi:7139.54	su:7038.54	yi:7001.08	yi:6805.40	yi:4005.60
su:3842.31	ba:3072.61	su:2984.71	su:2800.40	su:2250.10
ce:2509.41	ce:2206.91	ce:2109.10	ce:2000.97	ce:1807.50
na:2013.87	kuma:2001.01	kuma:1903.11	kuma:1730.20	kuma:1540.10
ne:1500.32	ne:1302.12	ne:1205.01	ne:1140.46	ne:1000.50
ke:1200.89	za:1150.75	za:1075.00	za:900.70	za:700.70
za:878.01	ke:798.41	ke:645.31	ke:570.05	ke:502.15
yan:623.11	daga:611.55	daga:599.01	daga:500.44	daga:425.10
suka:501.22	suka:499.92	suka:474.75	suka:416.20	suka:397.63
shi:453.07	shi:413.37	shi:401.05	shi:395.00	shi:380.60
wa:400.33	wa:395.23	wa:360.48	yan:340.10	wa:300.10
sun:398.11	sun:377.21	sun:327.06	wa:311.30	sun:278.20
ga:225.01	ga:205.15	ga:200.075	ga:190.05	yi:174.50
cikin:214.11	cikin:195.61	cikin:180.65	cikin:165.50	cikin:159.71
daga:200.22	yan:160.52	yan:156.20	sun:135.70	ga:120.40

The proportion of common parts were computed in the adjacent lists, and the list achieved saturation after fourth experiment, as illustrated in Table 4. The results illustrated the proportion of common parts using different words selection; top 25, top 50 and top 75. The best scores were obtained using the largest corpus comprising of 166813-187143 words.

The accuracy of the proportion increases with the increase of the corpus size, thus the larger the words in the documents the more accurate the proportion. Also, the accuracy of the list is affected by the number of stop-words in the lists, the lower the number the better the accuracy of the list, as shown in Figure 2. Finally total number of 81 words were selected for the list.

Table 4. Proportion of common words under corpora with different scales

Corpus size	40869-105436 words	105436-146601 words	146601-166813 words	166813-187143 words
Top 25	0.92	0.96	0.96	1
Top 50	0.88	0.94	0.96	0.96
Top 75	0.88	0.88	0.93	0.99

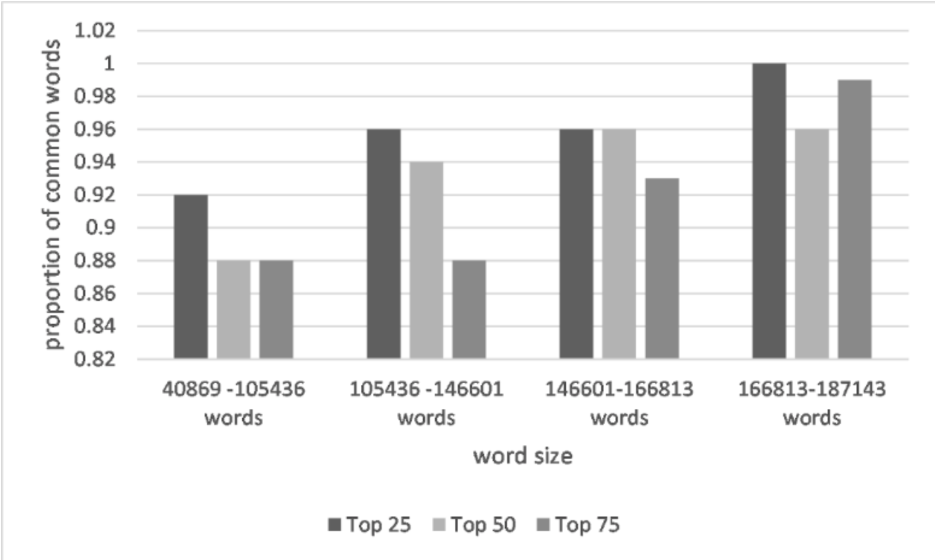

Figure 2. A bar chart for proportion of common words under corpora with different scales

3.1. Final list

The final list comprising of 81 distinct hausa stop words generated using the method described above are presented in Table 5. The list is created by selecting the words that produced by both the frequency and statistics methods. It comprised of most common hausa prepositions, conjunction pronouns.

Table 5. Final list of hausa stop words

List of hausa Stop Words							
da	wannan	yake	suka	daga	idan	abin	cikin
ya	wa	suke	sun	mai	yayin	ana	shi
ta	wanda	hakan	wasu	za	babu	in	bayan
na	an	hada	kan	cewa	baya	ita	ga
ba	wani	akan	ma	yan	tare	akwai	kai
kuma	sai	aka	kamar	ko	yadda	sake	amma
yi	masu	bai	tun	inda	don	irin	sa
ne	domin	mu	wadanda	samu	ake	tana	wajen
ce	dai	ke	su	yanzu	zai	ciki	har

4. CONCLUSION

Removing stop words is crucial in natural language processing and general artificial intelligence researches. Due to the non-availability of hausa stop words, this research filled the gap by creating a general list of hausa stop words using both frequency and statistics method. The list is created by selecting the words that appears using both methods. The total of 81 hausa words were finally selected after various consideration.

REFERENCES

[1] T. Y. Chong and R. E. Banchs, “An Empirical Evaluation of Stop Word Removal in Statistical Machine Translation,” *Computational Linguistics*. pp. 30–37, 2012, Accessed: Jan. 20, 2022. [Online]. Available: https://aclanthology.org/W12-0104.

[2] L. Hao and L. Hao, “Automatic identification of stop words in chinese text classification,” in *Proceedings - International Conference on Computer Science and Software Engineering, CSSE 2008*, vol. 1, 2008, pp. 718–722, doi: 10.1109/CSSE.2008.829.

[3] R. Rani and D. K. Lobiyal, "Automatic Construction of Generic Stop Words List for Hindi Text," *Procedia Computer Science*, vol. 132, pp. 362–370, 2018, doi: 10.1016/j.procs.2018.05.196.

[4] O. D. Tijani and S. A. Onashoga, "An Auto-Generated Approach of Stop Words using Aggregated Analysis," in *13th International Conference on Information Technology Innovation for Sustainable Development*, 2017, no. July.

[5] D. Na and C. Xu, "Automatically generation and evaluation of stop words list for Chinese patents," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 13, no. 4, pp. 1414–1421, Dec. 2015, doi: 10.12928/TELKOMNIKA.v13i4.2389.

[6] A. K. Pandey and T. J. Siddiqui, "Evaluating Effect of Stemming and Stop-word Removal on Hindi Text Retrieval," in *Proceedings of the First International Conference on Intelligent Human Computer Interaction*, Springer India, 2009, pp. 316–326.

[7] J. Atwan, M. Mohd, and G. Kanaan, "Enhanced Arabic Information Retrieval: Light Stemming and Stop Words," in *Communications in Computer and Information Science*, vol. 378 CCIS, Springer Berlin Heidelberg, 2013, pp. 219–228.

[8] B. Patel and D. Shah, "Significance of stop word elimination in meta search engine," in *2013 International Conference on Intelligent Systems and Signal Processing, ISSP 2013*, Mar. 2013, pp. 52–55, doi: 10.1109/ISSP.2013.6526873.

[9] M. Çağataylı and E. Çelebi, "The effect of stemming and stop-word-removal on automatic text classification in Turkish language," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9489, Springer International Publishing, 2015, pp. 168–176.

[10] I. Veritawati, I. Wasito, and T. Basaruddin, "Text preprocessing using annotated suffix tree with matching keyphrase," *International Journal of Electrical and Computer Engineering*, vol. 5, no. 3, pp. 409–420, Jun. 2015, doi: 10.11591/ijece.v5i3.pp409-420.

[11] B. R. Ganesh, D. Gupta, and T. Sasikala, "Grammar error detection tool for medical transcription using stop words parts-of-speech tags ngram based model," in *Advances in Intelligent Systems and Computing*, vol. 712, Springer Singapore, 2018, pp. 37–49.

[12] H. Li, Q. Chen, and X. Wang, "An improved method for semantic similarity calculation based on stop-words," in *Communications in Computer and Information Science*, vol. 481, Springer Berlin Heidelberg, 2014, pp. 339–347.

[13] D. Munková, M. Munk, and M. Vozár, "Influence of stop-words removal on sequence patterns identification within comparable corpora," in *Advances in Intelligent Systems and Computing*, vol. 231, Springer International Publishing, 2014, pp. 67–76.

[14] M. Jaiswal, S. Das, and Khushboo, "Detecting spam e-mails using stop word TF-IDF and stemming algorithm with Naïve Bayes classifier on the multicore GPU," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 4, pp. 3168–3175, Aug. 2021, doi: 10.11591/ijece.v11i4.pp3168-3175.

[15] N. N. Moon *et al.*, "Natural language processing based advanced method of unnecessary video detection," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 6, pp. 5411–5419, Dec. 2021, doi: 10.11591/ijece.v11i6.pp5411-5419.

[16] T. Islam, N. Ahmed, and S. Latif, "An evolutionary approach to comparative analysis of detecting bangla abusive text," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 2163–2169, Aug. 2021, doi: 10.11591/EEI.V10I4.3107.

[17] B. A. Benali, S. Mihi, I. El Bazi, and N. Laachfoubi, "New approach for Arabic named entity recognition on social media based on feature selection using genetic algorithm," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 2, pp. 1485–1497, Apr. 2021, doi: 10.11591/ijece.v11i2.pp1485-1497.

[18] M. Sadeghi and J. Vegas, "Automatic identification of light stop words for Persian information retrieval systems," *Journal of Information Science*, vol. 40, no. 4, pp. 476–487, Apr. 2014, doi: 10.1177/0165551514530655.

[19] S. Popova and G. Skitalinskaya, "Keyphrase extraction using extended list of stop words with automated updating of stop words list," in *Advances in Intelligent Systems and Computing*, vol. 689, Springer International Publishing, 2018, pp. 374–385.

[20] S. Girmaw and V. Khedkar, "Automatic Generation of Stopwords in the Amharic Text," *International Journal of Computer Applications*, vol. 180, no. 10, pp. 19–22, Jan. 2018, doi: 10.5120/ijca2018916161.

[21] D. Namly, K. Bouzoubaa, R. Tajmout, and A. Laadimi, "On Arabic Stop-Words: A Comprehensive List and a Dedicated Morphological Analyzer," in *Communications in Computer and Information Science*, vol. 1108, Springer International Publishing, 2019, pp. 149–163.

[22] K. Chekima and R. Alfred, "An automatic construction of Malay stop words based on aggregation method," in *Communications in Computer and Information Science*, vol. 652, Springer Singapore, 2016, pp. 180–189.

[23] W. Medhat, A. H. Yousef, and H. Korashy, "Corpora Preparation and Stopword List Generation for Arabic data in Social Network," Oct. 2014, Accessed: Jan. 20, 2022. [Online]. Available: http://arxiv.org/abs/1410.1135.

[24] A. Alajmi, E. M. Saad, and R. R. Darwish, "Toward an ARABIC Stop-Words List Generation," *International Journal of Computer Applications*, vol. 46, no. 8, pp. 8–13, 2012.

[25] P. Daowadung and Y. H. Chen, "Stop word in readability assessment of Thai text," in *Proceedings of the 12th IEEE International Conference on Advanced Learning Technologies, ICALT 2012*, Jul. 2012, pp. 497–499, doi: 10.1109/ICALT.2012.9.

[26] R. M. Rakholia and J. R. Saini, "A rule-based approach to identify stop words for Gujarati language," in *Advances in Intelligent Systems and Computing*, vol. 515, Springer Singapore, 2017, pp. 797–806.

[27] N. Khan, M. P. Bakht, M. J. Khan, A. Samad, and G. Sahar, "Spotting Urdu Stop Words by Zipf's Statistical Approach," Dec. 2019, doi: 10.1109/MACS48846.2019.9024817.

[28] J. K. Raulji and J. R. Saini, "Generating stopword list for sanskrit language," in *Proceedings - 7th IEEE International Advanced Computing Conference, IACC 2017*, Jan. 2017, pp. 799–802, doi: 10.1109/IACC.2017.0164.

[29] T. Asubiaro, "Entropy-Based Generic Stopwords List for Yoruba Texts," *Entropy*, vol. 02, no. 05, pp. 1065–1068, 2013, [Online]. Available: http://ijcit.com/archives/volume2/issue5/Paper020530.pdf.

[30] W. Medhat, A. Yousef, and H. Korashy, "Egyptian Dialect Stopword List Generation from Social Network Data," *The Egyptian Journal of Language Engineering*, vol. 2, no. 1, pp. 43–55, Apr. 2015, doi: 10.21608/ejle.2015.60258.

[31] M. A. Yaghoub-Zadeh-Fard, B. Minaei-Bidgoli, S. Rahmani, and S. Shahrivari, "PSWG: An automatic stop-word list generator for Persian information retrieval systems based on similarity function & POS information," in *Conference Proceedings of 2015 2nd International Conference on Knowledge-Based Engineering and Innovation, KBEI 2015*, Nov. 2016, pp. 111–117, doi: 10.1109/KBEI.2015.7436031.

[32] R. M. Rakholia and J. R. Saini, "Information retrieval for Gujarati language using cosine similarity based vector space model," in *Advances in Intelligent Systems and Computing*, vol. 516, Springer Singapore, 2017, pp. 1–9.

[33] J. F. Sohana, R. J. Rupa, and M. Rahman, "Bengali Stop Word Detection Using Different Machine Learning Algorithms," in *Algorithms for Intelligent Systems*, Springer Singapore, 2021, pp. 131–142.

[34] S. Sarica and J. Luo, "Stopwords in technical language processing," *PLoS ONE*, vol. 16, no. 8 August, p. e0254937, Aug. 2021, doi: 10.1371/journal.pone.0254937.

BIOGRAPHIES OF AUTHORS

Abdulkadir Abubakar Bichi is a graduate of Computer Science from the Kano University of Science and Technology Wudil. MSc Computer Science from Universiti Teklogi Malaysia (2013). He is currently a PhD candidate at School of Computing, Faculty of Engineering, Universiti Teklonogi Malysia. His research interests are in the field of information extraction, natural language processing, graph algorithms and machine learning algorithms. He is a charted member of Computer Practioners of Nigera (cpn) and a full member Nigerian Computer Society (ncs). He can be contacted at email: engraabubakar@gmail.com.

Ruhaidah Samsudin received the B.Sc., Msc. and PhD degree in computer science from the Universiti Teknologi Malaysia (UTM). She is currently a Senior lecturer at the School of Computing, Faculty of Engineering, Universiti Teklonogi Malaysia. Current research interests include soft computing techniques and applications, Time Series forecasting and applications, and hydrological science. She can be contacted at email: ruhaidah@utm.my.

Rohayanti Hassan received the B.Sc., Msc. and PhD degree in computer science from the Universiti Teknologi Malaysia (UTM). She is a senior lecturer at the School of Computing, Faculty of Engineering, Universiti Teklonogi Malaysia. Current research interests are: Artificial Intelligence, Biometrics, Software Engineering, Bioinformatics. She can be contacted at email: rohayanti@utm.my.