APPLICATION FOR TRANSLATING BATAK TOBA LANGUAGE TO INDONESIAN USING *BOYER MOORE* ALGORITHM BASED ON ANDROID

Stephani Silalahi*[1], Rakhmat Kurniawan R[2] , Aidil Halim Lubis[3]
[1,2,3] Computer Science Study Program, State Islamic University of North Sumatra, Medan, Indonesia
*e-mail: [1]*phanisilalahi@gmail.com,[2]rahkmat.kr@uinsu.ac.id,[3]aidilhalimlubis@uinsu.ac.id*

The lack of knowledge and the lack of preservation patterns as well as knowledge about the Toba Batak language in Indonesia causes the language to be less preserved, especially for young people who continue to follow the times. This is due to the process of globalization and urbanization which has the potential to trigger cultural assimilation and acculturation, especially in urban areas. This situation triggers the formation of a new language (slang) that is preferred, especially by young people, which unwittingly makes us lose our identity as a society that has its own tribes and customs and the emergence of national and international schools that require students to communicate using foreign languages. Therefore, it is necessary to have a more effective learning pattern so that the Toba Batak language is preserved. One of them is by creating a translator application from Toba Batak language into Indonesian by using the Android-based Boyer Moore algorithm. The system created has 100 Toba Batak vocabulary which can later be translated into Indonesian. This application is also made as one of the dissemination and introduction of the Toba Batak language to the outside community so that the Toba Batak language will be preserved because this translator application is able to translate from Toba Batak language into Indonesian or vice versa properly based on the applicable Toba-Indonesian Batak dictionary reference.

Keywords: Toba Batak Language, Indonesian Language, Boyer Moore Algorithm.

I. INTRODUCTION

Indonesia is the country with the second most languages in the world with 715 languages. North Sumatra Province has regional languages, one of which is the Toba Batak language. Toba Batak language is an indigenous language in the Samosir area which is used as a means of communication. Toba Batak language in its development has experienced a decline in the younger generation both in the speaking area and in the city. This is a concern of the author which can have a negative impact on the development of the Toba Batak language. Teaching the Toba Batak language that is not maximized and allowing the entry of Indonesian and foreign languages into the Toba Batak language is one of the problems faced by the development of the Toba Batak language [1].

According to the 2010 census, the Toba Batak language is spoken by 1,610,000 people, or only about 25.62% of the total Toba Batak ethnic population of 4,160,000. When viewed from the level of use, a number of areas with a majority of ethnic Toba Batak residents show a negative language attitude towards the Toba Batak language. The level of use of the Toba Batak language among residents in Narumonda VII Village, District. Siantar Narumonda, Kab. Toba only amounted to 10.5% [2]. While in Kwala Bekala Village, Medan Johor District, Medan City, the use of Toba Batak language is only 30% [3]. According to research by Marpaung & Tampubolon, 2022 also stated that this threat can also be caused because the younger generation is increasingly reluctant to use the Toba Batak language in social life. They tend to use Indonesian in communicating in their own area. The use of the Toba Batak language itself is due to the fact that teenagers do not have a view of their local language. This must be addressed immediately with efforts to generate willingness in regional languages, especially the Batak toba language in the hope that the extinction of the Batak toba language can be prevented.

Technological advances must be utilized in an effort to preserve regional languages, especially the Batak toba language. Artificial intelligence technology is one technology that can answer these challenges. Artificial intelligence is a technology that can adopt human intelligence to be implemented into computers. Translator application is one of the artificial intelligence technologies that can be used. The translator application has input in the form of text which then the text will be translated according to the desired language.

The translator application that will be built uses one of the methods, namely the string matching search method. String matching method is a string matching method used to search for a string called pattern in a string called text. One of the string matching methods that will be used is Boyer Moore algorithm. Boyer Moore algorithm is an approach that matches strings by looking from the right

side of the pattern then heading to the left by tracking characters in the text [4].

Boyer Moore algorithm starts matching characters from the right side of the pattern. The idea behind this algorithm is that by starting the matching characters from the right, rather than the left, more information will be gained. Boyer Moore algorithm includes the most efficient string matching algorithm compared to other string matching algorithms [5]. According to research Husni Rifqo, n.d 2019, The advantage of the Boyer Moore algorithm is that the longer the pattern to be searched, the shorter the search time will be. While the disadvantages of the Boyer Moore algorithm are that it is slower for short patterns and is not good for binnary string searches.

Based on previous research by Wicaksono et al. [6], entitled "Application of Boyer Moore Algorithm to Android-Based Biomedical Thymology Dictionary Application". In this study, the Boyer Moore Algorithm was applied to translate the meaning of language with word length and number of words in Biomedical Thymology which can only be translated into words not sentences. In doing this translation, what is sought is only in the form of words, therefore the words registered in the database will be searched with the string that the user needs. Where the meaning (description) of each word that will be searched will appear.

Other research conducted by Faqih et al. entitled "Application of String Matching Using Boyer Moore Algorithm in the Development of Online Book Search System". From this research, Boyer Moore's algorithm is used as a medium for processing book search websites, based on the results of the tests obtained, it shows that the program is feasible to use, because the associative formed and the resulting search value are the same as the output of the program developed.

II. LITERATURE REVIEW

A. Toba Batak language

Batak language is a language spoken by native speakers of the Toba Batak language. In this case, Toba Batak belongs to the Austronesian language family. Austronesian is the name of one of the language families that occupy the plains of Southeast Asia. The Austronesian family is divided into two branches, namely the Western Austronesian family (Indonesian or Malay) and the eastern Austronesian branch (Oceania languages in the geographical area of several islands located in the Pacific Ocean or Polynesian languages in the sub-region of Oceania located in the central Pacific Ocean [2][7][8].

B. Preservation of the Toba Batak Language

In the province of North Sumatra, Toba Batak is one of a group of Batak dialects. Batak grouping based on phonological comparison into northern Batak (including Karo, Alas and Dairi-Pakpak) and southern Batak which is divided into Simalungun dialect and Tobaic dialect (including Toba, Angkola and Mandailing). Afterwards, the Tobaic dialect split into Toba and a part consisting of Angkola and Mandailing [3].

C. Technology

Technology is a tool produced by humans to make it easier to carry out activities. Technology can be applied in all areas of life, including the field of communication. The use of technology can help to strengthen and maintain minority and endangered languages. Language plays an important role in the development of technology. The existence of technology afterwards creates several conveniences in the field of technology that make people's life patterns dependent on existing technological technology. From the word technology, which means the development and application of various equipment or certain systems used to solve a problem [9].

D. String Mathcing

String Matching is the process of finding all occurrences of a query called pattern into a longer string or text. Both strings consist of a set of characters called an alphabet denoted by $\sum$ (sigma). String matching is divided into two, namely exact matching and heuristic or statistical mathing [10].

E. Boyer Moore Algorithm

Boyer Moore algorithm is a string search algorithm. It was published by Robert S. Boyer, and J. Strother Moore in 1977. Boyer Moore's algorithm can be called an approach that matches strings by looking from the right then to the left by tracking characters in the text. This algorithm is widely applied to matching algorithms in many strings (mutil parttern). In the string search process, Boyer Moore algorithm reads the characters of the pattern from right to left. In cases where the number of characters in the pattern is less than the number of characters in the text, the algorithm uses precomputed functions. These two functions are Good-Suffix Shift and Bad-Character shift which are two shift functions used by Boyer Moore algorithm in taking search steps after an inequality is identified between the parttern character and the text character that performs the matching [6].

F. Search with Boyer Moore Algorithm

Systematically, the stages that the Boyer-Moore algorithm will perform when matching partterns are as follows:

1. Create a shift table for the searched pattern (P) using the Match Heuristic (MH) and Occurence Heuristic (OH) approaches, to determine the number of shifts to perform if a character is not matched in the text matching process (T).
2. If in the comparison process there is a mismatch between the character pair in the pattern and the text character, the shift is done by selecting one of the shift values from the two tables, and has the largest shift value from the Match Heuristic and Occurence Heuristic tables.
3. Two possible solutions in shifting the pattern, if the non-matching character is not in the pattern then the shift is as many as the number of characters in the pattern and if the non-matching character is in the pattern, then the number of shifts depends on the values in the Match Heuristic and Occurence Heuristic tables.

4. If the character in the text being compared matches the character in the pattern, then the position of the character in the pattern and the text is decreased by one position, then continue with matching at that position and so on.
5. If there is a mismatch between the pattern and text characters, then select the largest shift value from the Match Heuristic table and the Occurence Heuristic table value.
6. If all characters match, the pattern has been found in the text [11].

III. RESEARCH METHODS

Researchers interviewed one of the Batak traditional leaders in Tanjung Rejo Village, Percut Sei Tuan District. Researchers chose this village because of the lack of Batak toba language used by the residents of Tanjung Rejo Village and there has never been the same research on the subject of discussion about the use of Batak language in the village. In system design, the process stages will be carried out. The steps needed to achieve design goals can be seen in the waterfall model. This method is carried out with a systematic approach, starting from the planning stage, data collection, needs analysis, system design, system testing, and system implementation. Each step to be passed must be completed one by one or in the ari cannot jump to the next stage and run sequentially, therefore this langkat is called waterfall (Waterfall).

Boyer Moore Algorithm In performing a Boyer Moore search, the initial process is to place the window of the pattern in the available text. The search process starts from the rightmost character of the pattern. Each character will be compared one by one. If there is a mismatch, the possible shift values will be checked in the good suffix shift and bad-character shift character shift tables. The largest value obtained will be taken and the window shift will be performed according to that value.

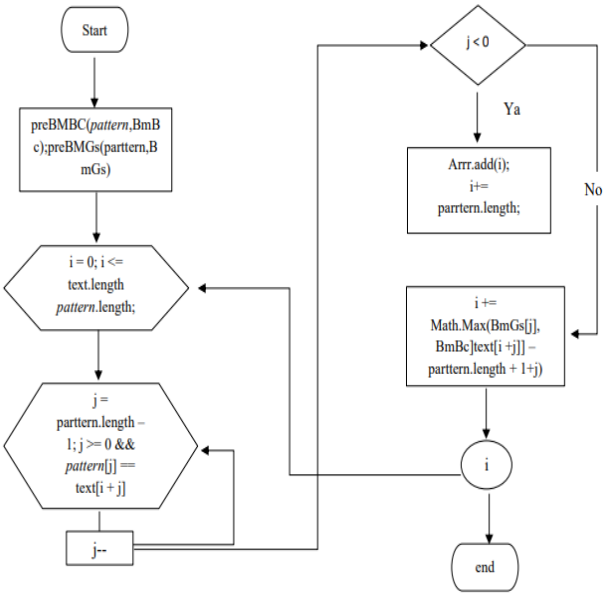


Figure 1. *Boyer Moore* Algorithm Flowchart

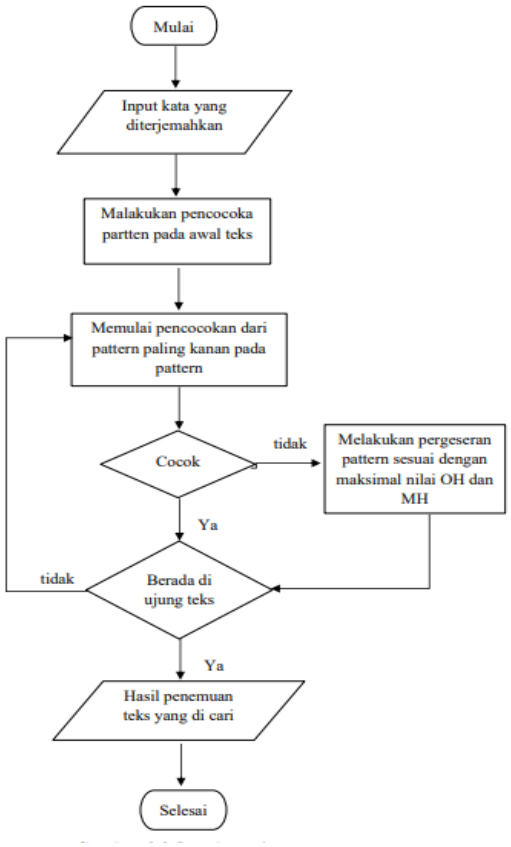


Figure 2. System Flowchart

IV. RESULTS AND DISCUSSION

A. *Data Analysis*

Analysis is a step of understanding the problems to be solved before taking action or decisions in designing the system to be made. The data used in this study is in the form of data on the translation of Batak toba language to Indonesian and vice versa which is taken based on the Toba Batak language dictionary. Where in the application that will be developed will be used as many as 1000 vocabulary that will later be translated.

B. *Data Representation*

The process of translating the Toba Batak language into Indonesian requires a Boyer Moore algorithm calculation stage to get character matches based on the word being searched:

C. *Implementation of Boyer Moore algorithm*

The search data that will be used is a sample table of words that will be searched manually in the following table.

Table 1. Sample Word Database

No.	Says
1	Eat
2	Home
3	Sleep
4	Amang
5	Didia

In table 4.2 is a sample of available words, then the user will enter the word to be translated in the translation field. The word will be matched with the word in the database using the Boyer Moore algorithm.

D. PreBmBc procedure

The preBmBc procedure has three important values, including:

1. Pattern, as the subject of matching against the text.
2. Character, as the characters contained in the pattern.
3. Occurence Heuristic (OH), as the shift value obtained when finding a character mismatch.
4. Shift, as the value achieved when shifting from the right to the left of the pattern.

E. Steps of procedure implementation:

1. Create or assign an empty value to the BmBc stack.
2. Calculate the length of the pattern. If the length is not more than one, then stop the process by adding the OH value and character directly into the BmBc stack. Otherwise, proceed to the next process.
3. The pattern character enumeration starts from the 2nd rightmost character.
4. Compare each enumerated character against the stack BmBc, if the enumerated character is not found in the stack, then add the character to the stack where OH is equal to the number of character shifts that have been performed.
5. Perform step-4 again, moving 1 character to the left until you reach the leftmost character continuously.
6. If it has reached the leftmost character, enumerate the rightmost character and return to step-4.

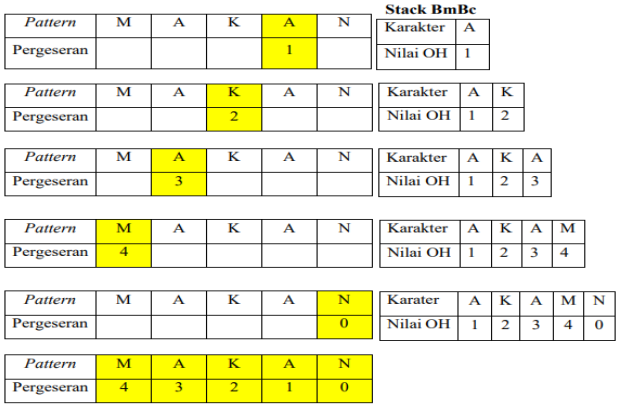
The first case of finding the word EAT in the BmBc process: *Pattern*: EAT
Completion:

Table 2. index on the eating pattern

Index	0	1	2	3	4
Pattern	M	A	K	A	N

Do the calculation, OH = (*length* - index -1)
length = character length = 5

1. The first character is "M" with Index = 0
   OH = (5 - 0 - 1 = 4) then character value "M" = 4
2. The second character is "A" with index = 1
   OH = (5 - 1 - 1 = 3) then character value "A" = 3
3. The third character is "K" with index = 2
   OH = (5 - 2 - 1= 2) then character value "K" = 2
4. The fourth character is "A" with index = 3
   OH = (5 - 3 - 1 = 1) then character value "A" = 1
5. The fifth character is "N" with index = 4
   OH = (5 - 4 - 1 = 0) then character value "N" = 0



Case completion of the preBmBc procedure for the word eat

In the figure above, the designated character will be added to the BmBc stack if it is not found on the Bmbc stack with an OH value that corresponds to the shift value.

a. PreBmGs procedure

The preBmGs procedure has six important values, including:

1. *Pattern*, as the subject of matching against the text.
2. *Match Heuristic* (MH), as the shift value obtained when finding *suffix* matches.
3. *Compare*, as the suffix or the number of characters to the right of a character *pattern* obtained from a right-to-left shift.
4. *Prefix*, as a prefix or *pattern* character obtained from a left-to-right shift.
5. *Suffix*, as the suffix to the right of the *prefix*.
6. Shift, as the value achieved when shifting from *compare*.

Steps for implementing the procedure:

1. Create or assign empty values to the BmGs and *Compare* stacks.
2. Perform the same action with preBmBc if the *pattern* length is not more than one.
3. Enumerate the *compare* against the *pattern*. Then save the enumeration result by adding it to the *compare* stack.
4. Enumerate the *prefix* and *suffix* against the *pattern*. Then store the result of *prefix* enumeration into the BmGs stack, and compare the *suffix* against *compare* by looking for matches between each *prefix*.
5. If the lengths of the two *suffixes* are not equal, count the one with the largest length to the right along the *suffix* with the smallest length. If a match is found between *suffixes*, compare the *prefixes of the* two *suffixes*. If the compared prefixes are the same, then the match is not acceptable.
6. For *compare*, the *prefix is* obtained when enumeration occurs in *compare*, the *prefix* will be equal to the rightmost character of the enumeration result against *compare* before enumeration. If no enumeration occurs, then *prefix-compare is* null.
7. The index of the rightmost *prefix*, the MH value must always be 1. While the MH value at the other *prefixes*, is given when a mismatch is found between the two *suffixes* by the value of the shift formed *compare*
8. Repeat step 5 until the last *compare*.
9. Repeat step 5 until the last *suffix*.

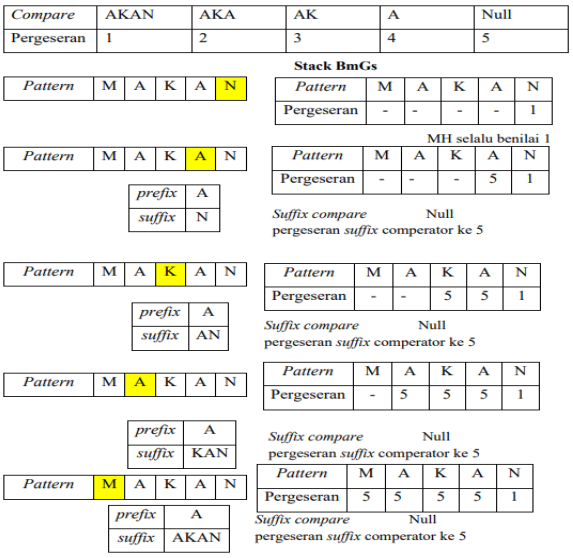
The first case of the EAT word search in the BmGs process:
Pattern: EAT
Solution:

Table 3. *Prefix* and *suffix* enumeration list of *patterns*

Suffix (right-left)	*Prefix*	*Suffix* (left-right)
Null	M	WILL
A	A	AKA
AK	K	AK
AKA	A	A
WILL	N	Null

From the table above, there has been a cut on both sides of the *suffix* against the *prefix* in the pattern. It can be seen that each *suffix* is united by the *prefix* until it

becomes a whole *pattern*. While null means empty, that is, all matching conditions will be true if a comparison is made to the character.



Case Completion Procedure of preBmGs of the word eat

In the above solution, it can be seen that the shift in the *pattern* always follows the decision of the OH and MH comparison results. In the case of finding the word eat on shift 1 the value obtained from the comparison between OH and MH is 1 step, then the match is shifted by 1 step. In shift 2, the largest value is taken from the comparison of OH and MH by 2 steps, after shifting each text character and *pattern* match, then the shift is complete. In general, standard matching will match character by character, when a mismatch is found, the *pattern* will be shifted one step in the next direction. But this is not the case with *Boyer Moore* algorithm. This is the main reason why the author applies this algorithm as a word translator algorithm in the Batak toba language to Indonesian when string matching is needed.

F. System Implementation

The following will explain the display of the results of the Toba Batak to Indonesian language translator application using the andoroid-based *Boyer Moore* algorithm which can be seen as follows:

1. Main view (splash)

The splash screen display can be seen in Figure 4.20 The splash screen is the display that first appears before the start page:

Figure 5. Splash Screen Display When the Application is Clicked

2. App Home Page

The initial page of the application can be seen in Figure 6. The initial page will be displayed after the splash screen is complete, after which the initial page of the application will display several selection buttons, namely Home, Toba Batak Translator - Indonesia, Indonesian translator - Toba Batak, about and exit.

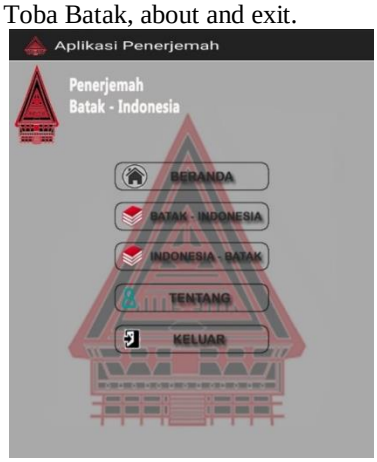


Figure 6. Home page view of the application

3. Home Page

The home page display can be seen in Figure 4.22 when the home menu is clicked by the user, the application will display information about a brief explanation of the translator application.

Figure 7. Home Page Display

4. Toba Batak page - Indonesia

The display of the Toba Batak - Indonesia page can be seen in Figure 4.23 as follows.

Figure 8. Display of the Toba Batak - Indonesia Page

5. Indonesian Page - Batak Toba

The display of the Indonesia - Batak Toba page can be seen in Figure 4.24 as follows.

Figure 9. Indonesia - Batak Toba Page Display

6. About App Page

The page about the application can be seen in Figure 4.25 The page about the application is a page that contains brief information about the author.

Figure 10. About Display Page

V. CONCLUSIONS AND SUGGESTIONS

From the final project research conducted, the following conclusions can be drawn: Based on this research, the words that can be tested are 1000 words in the database. In finding words using Boyer Moore algorithm, it is important to find the value of BmBc and BmGs where the value determines the shift of each word to be searched. Based on the results of testing the process of translating Toba Batak language into Indonesian using the Boyer Moore algorithm, the words searched are in accordance with the words in the database. The android-based language translator system is designed using Eclipse by using the Boyer Moore algorithm in a mobile manner so that it can help users when using the system anywhere without using a network connection.

The suggestions given by the author for further development and improvement of this system are as follows: It is hoped that in the future there will be good development in the form of adding application features that are accordingly needed such as copy, cut, text to speech, and other additional features. In the application of the Toba Batak language translator into Indonesian language to be developed again in terms of adding information or information on the word being searched. To be able to use more effective language on vocabulary.

REFERENCES

[1] T. Sibarani, "Pelestarian Bahasa Batak Toba Dari Tinjauan Sosiologi Dan Struktur Bahasa". Medan Makna: Jurnal Ilmu Kebahasaan Dan Kesastraan, 13(2), 203–314. 2019.

[2] D. K. Marpaung, & F. Tampubolon, "Pemertahanan Bahasa Batak Toba Di Desa Narumonda Vii Kecamatan Siantar Narumonda Kabupaten Toba". Journal Of Language Development And Linguistics, 1(1), 29–44. 2022.

[3] M. F. Nainggolan, & M. L. Girsang, M. L. "The Analysis Of Language Maintenance Of Batak Toba And Karo Language". Anglo-Saxon: Journal Of The English Language Education Study Program, 10(2), 95–104. 2019.

[4] I. Ahmad, R. I. Borman, G. G. Caksana, & J. Fakhrurozi, "Implementasi String Matching Dengan Algoritma Boyer-Moore Untuk Menentukan Tingkat Kemiripan Pada Pengajuan Judul Skripsi/Ta Mahasiswa (Studi Kasus: Universitas Xyz)". Sintech (Science And Information Technology) Journal, 4(1), 53–58. 2021.

[5] C. Irawan, & M. Riyan Pratama, "Perbandingan Algoritma Boyer-Moore Dan Brute Force Pada Pencarian Kamus Besar Bahasa Indonesia Berbasis Android". Bios: Jurnal Teknologi Informasi Dan Rekayasa Komputer, 1(2). 2020.

[6] I. B. Wicaksono, I. H. Santi, & F. Febrinita, "Penerapan Algoritma Boyer-Moore Terhadap Aplikasi Kamus Teminologi Biomedis Berbasis Android". In Jurnal Mahasiswa Teknik Informatika) (Vol. 6, Issue 2). 2022.

[7] L. H. Napitupulu. "The Correspondence Of The Proto-Austronesian Consonant Phonemes In Toba Batak Language: Korespondensi Fonem Konsonan Proto-Austronesia Pada Bahasa Batak Toba". Jurnal Bahasa Indonesia Prima (Jbip), 3(2), 311–318. 2021.

[8] B. Pranata, Y. Laia, & M. L. Gaol, "Perancangan Sistem Penyusunan Marga Suku Batak Toba Berbasis Web". Jurnal Sistem Informasi Dan Ilmu Komputer Prima (Jusikom Prima), 3(1), 17–23. 2019.

[9] R. F. Daud, "Dampak Perkembangan Teknologi Komunikasi Terhadap Bahasa Indonesia". Jurnal Interaksi: Jurnal Ilmu Komunikasi, 5(2), 252–269. 2021.

[10] S. R. Siregar. "Penerapan Algoritma Boyer Moore Pada Aplikasi Kumpulan Cerita Motivasi". Bulletin Of Multi-Disciplinary Science And Applied Technology, 1(1), 1–6. 2021.

[11] B. Harpad, S. Widya Cipta Dharma, N. J. M. Yamin, & S. Sur-El, "Penerapan Algoritma Boyer-Moore Dalam Perpustakaan Virtual Smk Muhammadiyah 1 Samarinda Sebagai Pendukung Pembelajaran Daring". Jurnal Ilmiah Matrik, 23(2). 2021..