International Journal of Computer Trends and Technology
ISSN: 2231 – 2803 / https://doi.org/10.14445/22312803/IJCTT-V69I2P114

Volume 69 Issue 2, 93-99, February 2021

Original Article

JavAnote: Tool and Strategies to Annotate Javanese Characters

Lucia D. Krisnawati[1], Aditya W. Mahastama[2]

[1,2] *Informatics Department, Universitas Kristen Duta Wacana Jl. Dr. Wahidin 5-25, Jogjakarta, Indonesia.*

Received Date: 08 January 2021
Revised Date: 25 February 2021
Accepted Date: 27 February 2021

Abstract *- Developing an automated annotation tool for an orthography that is no longer in use is challenging. The fact that the expertise on such orthography is quite rare will greatly affect the quality of the annotation. The targeted users that drive the User Interface design of the tool should also be carefully defined. In this paper, we describe a shift of our perspective on the functionality of an annotation tool that led us to deploy a dual function tool, i.e., a tool that on its front end serves as e-learning for Javanese characters and the annotation tool on its backend. As an e-learning tool, it supports achieving the first two cognitive learning objectives of Bloom's taxonomy by providing a drop-down list of tuples of Javanese characters and their transliteration as labels. Learners' engagement is enhanced by the drag and drop feature. Being run and used in a real environment, the tool has produced 15.414 valid annotated character segments*

Keywords *- Javanese character, Automated annotation tool, e-learning tool, Dual-function tool, OCR*

I. INTRODUCTION

The availability of annotated characters plays a crucial part in an optical character recognition (OCR) system whose task is to recognize either handwritten or printed texts. Prior to the annotation process, the characters as training data should be collected. The data collection could turn out as an intricate problem since it should represent its anticipated use case [1] and contribute to the success of the recognition process [2]. Even though the data annotation process is time-consuming and costly, it 'allows for greater accessibility of research objects' [1] since the annotation provides beneficial information to use properly in different schemes. Due to a large number of data, it is impossible to apply a bare-hand annotation scheme. Therefore, an annotation tool is desperately required to optimize and standardize this process.

This article describes the strategies, design, and properties of JavAnote, a tool for annotating Javanese characters. JavAnote is a part of *Trawaca* project whose aim is to preserve and digitize manuscripts written in Javanese characters using OCR technology. The tool is supposed to provide labeled characters for training data in the OCR project for Javanese characters. The consolidated project description and the annotation tools are available at https://trawaca.id/.

The paper is organized as follows: the next section briefly summarizes the previous work on the annotation tool, then followed by a discussion on the method and strategies used to build JavAnote. The functionalities of JavAnote are described in section 3. Section 4 will shortly describe its usage in a real environment. The discussion on the annotation results and JavAnote future perspective will end this paper.

II. RELATED WORK

Based on our literature survey, reports on text annotation tools are more dominant and developed than ones for characters, specifically for non-Latin characters. The text annotation tools are mostly web-based ([3]-[4]) to accommodate the multi-user, multi-label document annotation [5], or collaborative annotation scheme [6]. A different scheme is presented by a text annotation tool coined as SLATE. Designed to overcome the burden of installing and configuring an application, SLATE is a desktop-based application whose annotation is done in a console [7]. Thus, it was specifically designed for annotators who have been enough expertise in using computers. SLATE also supports annotation of characters, tokens, lines, or documents of different types such as free texts, labels, or links [7].

In the case of character annotation, an offline Chinese annotation tool that combined manual and automatic annotation strategies was devised [8]. The character labels were acquired by rewriting manually the same characters, line-by-line, of a page on specific handwritten texts. The task of the annotation tool is to segment each character on both the handwritten texts and their copies. Then, it aligns the outputted character segments based on the index and uses the isolated characters from the copied version as labels for characters from the handwritten texts. The tool performance

was improved by accommodating the online character annotation process, which gathered the spatial and temporal information [9].

A semi-automatic tool for annotating Arabic online handwritten texts is described by Elanwar et al. [10]. The tool provides utilities for segmenting and annotating handwritten Arabic characters. The labeled characters are then feeding as training data of a character recognizer. The semi-automatic here refers to the user-friendly interfaces which provide options to perform automatic or manual annotation [10].

III. A CASE OF JAVANESE CHARACTERS

Though Javanese is regarded as a classical language with literary tradition over a thousand years [11], its orthography is not used anymore in modern daily life. The introduction of the Latin alphabet during the Dutch colony in the 19th century and the ban on using native scripts during the Japanese occupation gradually diminished its usage [12]. After Indonesian independence, the tendency to use Indonesian written in Latin was greater, and it made Javanese script which is much more complicated has been abandoned. Nowadays, Javanese script is learned at schools, used exclusively by scholars, and for decoration, such as being written on the street signs under the Latin alphabet only in some regions of Java [13]. However, Javanese is still spoken by 68.3 million speakers [14] and appears in the Latin alphabet.

The Javanese script is classified as Abugida, a segmental writing system in which consonant-vowel sequences are written as a syllable unit with an inherent vocal 'a' [2]. There are 20 basic consonant-vowel sequences called Carakan, 20 consonant clusters, 20 for capital sequences, 5 for capital vocals, 16 punctuations, and 10 for loan-syllables. In addition, there are 19 characters that could be considered as diacritics representing vowels other than 'a', onset consonant clusters, and coda consonants.

IV. STRATEGIES AND METHODS

Developing an annotation tool for Javanese characters which are no longer in use is very challenging. The rationale is that firstly the labels used to annotate cannot simply take the same form as in [8], which uses the same orthography. Secondly, the low literacy rate and rare expertise on these characters may influence the quality of the annotation. The third problem deals with the tricky way of solving the long-running annotation process to get a sufficient number of samples in each character class.

To cope with these problems, we came up with an idea to crowdsource the annotation process. However, we encountered that this method did not solve our problems. Then we changed our perspective on the tool functionality and who could be the annotators (users). As its consequence, we have developed two versions of annotation tools which we coined as JavAnote 1 and JavAnote 2. Number one and 2 refers to the versions of the tool.

A. Software development life cycle

Our former perspective on an annotation tool was that it has a single function, i.e., to automate the annotation process. Based on this perspective, we identified three problems mentioned at the beginning of section 2. To cope with the first problem, it was decided to use the Latin transliteration of each Javanese character sequence as its label. The label in Latin was also aimed to deal with the annotator's browser problems of being unsupportive to write and display the whole range of Javanese characters.

The next step was to determine the source documents for labeling. 3 books printed in a different era were chosen as samples of different fonts, character richness, and complexity. *Serat Mangkunegaran IV* part I, printed in 1853, was chosen to represent the Javanese character variety used till 19th century; the book Rome from the Bible was meant to capture the Javanese characters used in the early 20th century, and a Javanese lesson book printed in the 1970s was selected to represent the simplified modern Javanese characters.

In the development phase, the decision to crowdsource the annotation process led us to design a web-based tool that supports collaboration and interaction among annotators. The tool script was written in PHP 7, while its web interface was coded in CSS, Javascript, and jQuery AJAX. The CSS framework, Bootstrap, was also applied to accommodate access from and tool responsiveness to mobile devices. The completed annotation tool was then coined as JavAnote 1.

The testing - improvement cycle of JavAnote 1 was performed while it was used to annotate characters by a group of limited annotators from the Javanese Wikipedia Community. After a round of annotation processes, the annotators were asked to report the encountered bugs or provide feedback to improve JavAnote performance and usability. After four rounds of usage and improvement, JavAnote 1 was deployed for a crowdsourcing scheme. The description of the development life cycle of JavAnote was displayed in Figure 1.

The evaluation of the annotation results showed that the annotated characters were unequally distributed to character classes. This means that most character classes had an insufficient number of samples. Hence the annotated characters have not met the need of being training data in an OCR system. To rerun a crowdsourcing annotation along with its evaluation process was time-consuming and would end up the same thing. This situation led us to the perspective that an annotation tool could serve more than one function

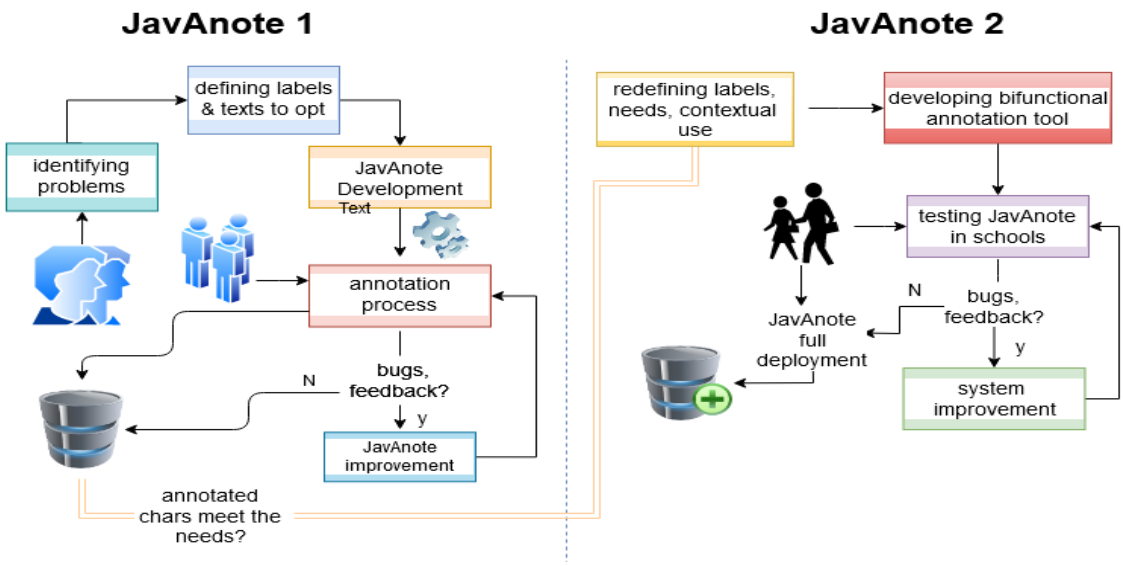

Fig. 1 A description of Java note development life cycle

With this perspective, we redefined JavAnote 1 functionalities. On the one hand, it functions as an e-learning tool -- an alternative teaching method in Javanese classes. On the other hand, it retains its function as an annotation tool. This means the labels, labeling system, database, and user interface should be redesigned. The next steps of developing JavAnote 2 are similar to JavAnote 1. The only difference is set on the location of testing it, i.e., in school labs(see Fig. 1).

B. Identifying users roles

As JavAnote 1 was used in a crowdsourcing way, we identified 3 groups of user roles, namely annotators, evaluators, and system manager, whose roles and requirement are as follows:

- **Annotators** are those who register and pass the literacy test of Javanese characters with an accuracy rate of character identification greater than 80%. Designed as collaborative annotation tools, JavAnote 1 accommodates annotators' needs for choosing any page and specific line of the book, annotating precisely, revising their annotation, and knowing the total number of characters that have been annotated. However, if an annotator is unable to finish annotating all characters in a line, another annotator should be able to finish it.
- **Evaluators** are those with a degree in or students of Javanese Studies. Most of them are members of the Javanese Wikipedia Community. Their tasks are to assess the accuracy of annotated labels, edit the wrong labels or elements of a character sequence identified by annotators. They are also able to do the annotation.
- **System managers** are the researcher team who should get all information regarding the results of the annotation process. They are also responsible for adding the system functionality, improving and redesigning the annotation tool based on the defined needs.

As an e-learning tool, JavAnote 2 has different user roles and specifications but has 3 groups of users as well. The role and specification of the system manager remain the same. However, the role of users and evaluators are replaced with students and teachers.

- **Students** who replace the annotator role should be able to identify an isolated compound-character in its context by writing its transliteration in Latin as an exercise of their Javanese lesson. Their work should be graded, and feedback on which parts they make mistakes in should also be provided.
- **Teachers** whose role replaces evaluators should be able to evaluate, assess, and grade the work of their students. They should also be able to show the correction of mistakes as a form of feedback to the process of learning the Javanese characters.

V. JAVANOTE FUNCTIONALITIES

The user roles defined in the former section have tacitly specified system requirements that have been implemented and reflected on its user interface and functionalities. Both JavAnote 1 and 2 supports the functionalities explained in the following subsections.

A. Multi-user

The multi-user feature, which enables both JavAnote versions 1 and 2 to be used simultaneously, was constructed by setting up session management with AJAX. To support this, each user role should register themselves and log in to the system in order to start a session of annotation. In JavAnote 1, passwords for register were exclusively generated and given to annotators and evaluators to avoid anyone signing in the system and doing trial and error annotation. In JavAnote 2, this strategy is retained for teachers, while anyone can theoretically register as a student as long as he/she has a high school student ID.

B. Collaborative annotation scheme

This scheme is communicated through various elements of Graphical User Interface (GUI) Design. One of its examples is demonstrated in Figure 2. In JavAnote 1, annotators are free to choose a page of a manuscript that directs them to its annotation site consisting of ± 250 segmented characters, as shown in Figure 2a. A character segment that has been annotated was marked with a yellow background color and white for one that has not been. By seeing the background color, another annotator is able to choose which character he/she should annotate. Besides, the button color and its label also highlight the collaborative feature.

In JavAnote 2, the collaboration is conveyed by buttons (See Figure 2b no. 3) on a page where students are free to choose a line of characters as their own exercise. The buttons have 4 labels: 'choose,' 'on-going,' 'continue,' and 'finished.' The Choose Button informs that the whole characters of this line have not been annotated, so any student can choose this line as her/his exercise. The On-going and Finished Buttons show the current situation of the task, and the buttons were disabled to click. The Continue Button informs that characters in that task have been partially annotated; hence other students can opt for this as their assignment to continue the annotation process.

C. Interactive Features

The interactive feature is strongly depicted on the annotation sites of both tools. In JavAnote 1, when an annotator clicks a button on each character segment (Fig. 2a), he will be directed to an annotation page, as shown in Figure 3. There, he needs to fill in the transliteration field only. The rests are provided as drop-down lists, which enable him to opt for the right meta-information and labels for a character segment. If the character segment comprises more than one element, he should click Add Button, which enables him to re-click the drop-down Category List, which provides Javanese character categories such as basic character, consonant cluster builder, or capital vocals. The drop-down Element List will provide only characters classified in a previously opted category. When all elements have been labeled, he needs to click the Save Button in green to send his annotation to the database server.

(a)

(b)

Fig. 2 The User Interface design, which supports collaborative and interactive annotation process in JavAnote 1 (a) and in JavAnote 2 (b).

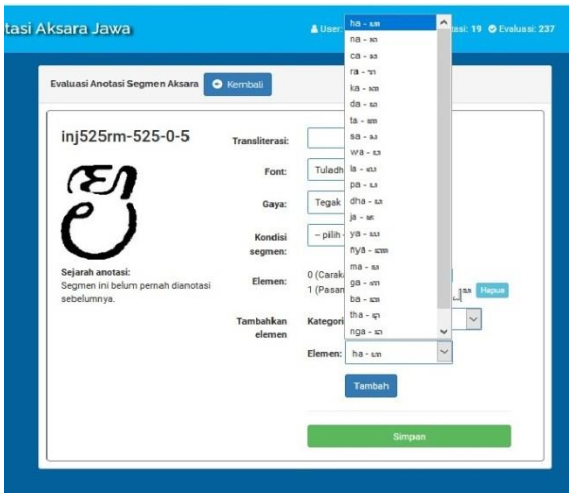

Fig. 3 The annotation page of JavAnote version 1

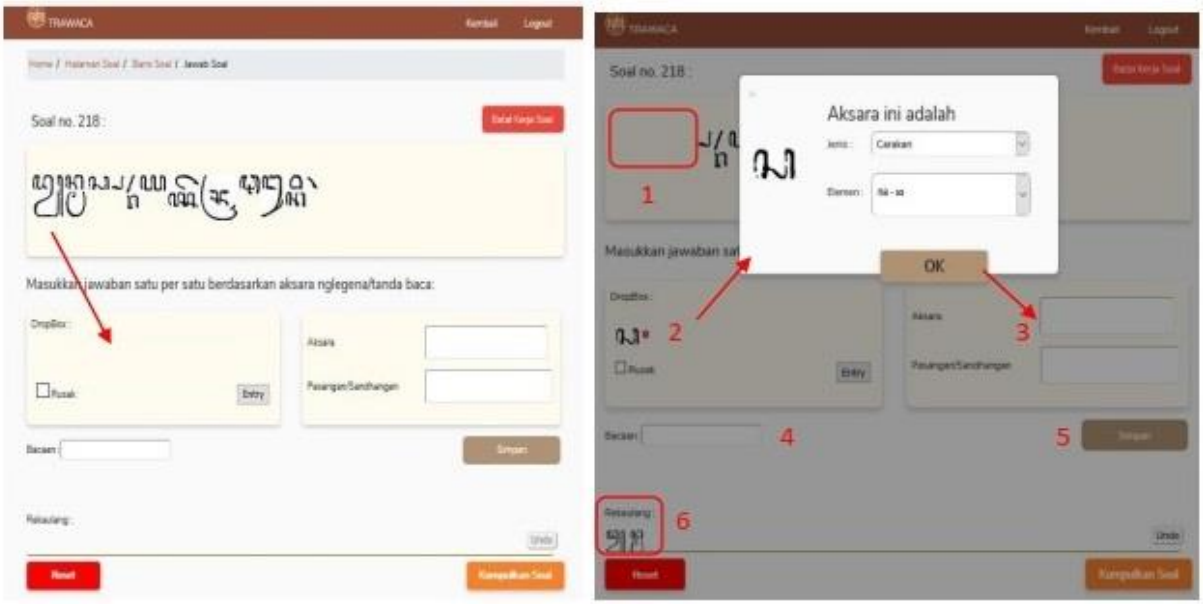

Fig. 4 The annotation page of javAnote 2 functioning as an exercise for students

Designed as a learning tool, the labeling in JavAnote 2 is done by dragging a character segment to a box provided below the task box (see Figure 4 left). The character segment is treated as an object, so when it is dragged and landed on the dropbox (Fig. 4 right no 2), it triggers a pop-up window displaying the character and drop-down lists for labeling. In order to do the exercise A.K.An annotation, students need to opt for a category that is provided in a *Category* drop-down list. After choosing the right character category, the *Element* drop-down list will display only characters included in that category, as shown in Figure 3. Completing this task, a student or learner should click the *OK* Button, which displays the results of his work on the temporary box (Fig. 4 no 3).

To give additional labels on a character segment having more than one element, the *Entry* button on the dropbox needs to click, which will redisplay the annotation pop-up window. After all elements of a character segment have been identified and labeled, the transliteration field (Fig. 4 no 4) needs to fill in. The interactive feature is shown by the Save (Fig 4 no 5), Undo, and Reset Buttons. The *Save* Button will display the temporary results of the annotation (Fig. 4 no 6). If a student remembers that it does not look like the character he is working on, he needs to click the *Undo* Button, which removes only the last character from the temporary list of display and restores it on its task box (Fig. 4 no 1). The *Reset* Button is used if a student is reluctant to submit their work, instead prefer deleting all characters that have been labeled.

The interaction between students and their teachers are also mediated on their dashboards. After a student log in JavAnote 2, he has a dashboard consisting of a menu to opt whether he needs to do an exercise or to look at his grade. A grade page displays the exercises done by students, and its status whether it has been corrected or graded by his teacher. If the student made a mistake and the teacher gave correction as feedback, then the correction is displayed above his/her work.

D. Dual function tool

Designed as an eLearning tool, The GUI of JavAnote version 2 emphasizes more on a learning process. The drag and drop feature is a technique to provide fun and support student engagement in learning Javanese characters [15]. The dragged character has a small red circle with a cross, upon which a click will return the character back to its former place (see Fig. 4 no 1-2). Besides, there are *Undo* and *Reset* Buttons that support learners' cognitive process when they realize they have made mistakes. The *Undo* Buttons will reset only the hindmost character in the range of annotated characters, while the *Reset* One will reset the whole characters (Fig 4 no 6) so that they are able to redo their exercises.

Learner cognition in remembering the Javanese characters is also carried through the drop-down *Element* List, which displays a tuple of labels in Latin and its Javanese characters, which has similarity with the drop-down list displayed in Fig. 3. By doing the exercise, learners repetitively choose the character on the list, and it automatically improves their memory on the Javanese character forms as they read their transliterations. Thus, Javanote 2 supports achieving, at least, the first two cognitive learning objectives of Bloom's taxonomy [16], i.e., knowledge and comprehension by remembering and understanding.

As an annotation tool, the student works are saved in a relational database whose table is related to the correction made by teachers. The labels (students' works) which are

evaluated, corrected, and saved by teachers are then considered as the annotated characters.

VI. USAGE AND RESULTS

Both versions of JavAnote have been deployed and used in a real environment. The JavAnote 1 has been used in one round of crowd-sourced annotation process and 4 rounds of evaluation. The crowd-sourcing process involved 36 active annotators whose number of annotated characters ranges from 1 to 3.053 characters for each annotator. The characters inputted to the system were taken from 72 pages of scanned texts mentioned before and resulted in 19.112 characters in total. However, the automated segmentation produced also over segmented as well as under-segmented characters, which we excluded from our data though they were also annotated by the annotators.

The evaluation process of the annotated characters was done to ensure that the labels which function as the ground truth data are correct and fulfill the requirement of being the gold-standard data. At first, the evaluation was done semi-manually through the provided Graphical User Interface for the evaluator, which checks each character on a page. However, this technique was time-consuming since there were only 21 active evaluators. To overcome the problem of time, we devised a code that groups the labels and their characters into classes and displays these classes along with their instance objects through GUI. Then the evaluator checks whether there are misclassified characters due to their annotated labels. The correction was then performed to each instance, which is mislabelled. This has reduced the time needed for performing the evaluation.

Table 1.
The Statistics on the annotation results using both versions of javanote

Item	Total number
Scanned pages	72
Character segments	19.112
Annotated characters	15.414
Evaluated characters	15.414
Undersegmented characters (broken)	3512
Oversegmented characters (merged)	309

Unlike JavAnote1, JavAnote 2 has been applied as a teaching tool in 4 classes of Javanese lessons and for a session of training for Javanese teachers. Designed as a web-based application, JavAnote 2 was used in school labs to make sure that each computer has been installed the Javanese font. Unluckily, the Covid-19 pandemic has slowed down the full operation of JavAnote 2 as an eLearning tool. The annotation results for the deployment of both versions of JavAnote are presented in Table 1. In total there have been 15.414 character segments have been annotated and evaluated.

VII. CONCLUSION

The two most important challenges in conducting annotation for an orthography that is no longer in use are the rare expertise and the time-consuming annotation process. In addressing these challenges, we came to an idea to devise a dual function tool, i.e., an annotation tool that on the surface serves as an eLearning for Javanese characters. The labeling is done as a part of exercises on identifying Javanese characters. The labeled characters that have been corrected and marked by teachers are included in the gold-label data. Being run and used in the real environment, both versions of our annotation tool have produced 15.414 valid annotated character segments.

One distinctive feature in our dual-function annotation tool is the drag and drop, which treats characters as image object movable by mouse. This feature is fit for a web-based application accessed by portable computers and PC. Unfortunately, it does not work for a mobile application. In the future, we would like to develop a feature resembling the drag and drop that is applicable to mobile devices such as a smartphone. We would also like to examine the effect of using the IT-based tool in learning Javanese characters, how far it helps learners to deconstruct the misconception that learning Javanese characters are boring and out-dated.

ACKNOWLEDGMENT

We would like to give our deepest gratitude to Wikimedia Indonesia (WMID), which has funded the OCR project and partially funded the JavAnote construction. We also give our thanks to the Javanese Wikipedia Community for their contribution to evaluating the character segments. Last but not least, we would also like to show our gratitude to the Informatics Department, Universitas Kristen Duta Wacana, which partially funded the construction of JavAnote version 2.

REFERENCES

[1] A. Barbaresi, Ad hoc and general-purpose corpus construction from web sources. Linguistics. ENS Lyon, English, (2015).

[2] L.D. Krisnawati, and A.W. Mahastama, Building classifier models for on-off Javanese character recognition, in M. Indrawan-Santiago et al. (eds.) iiWASProceedings,. New York: ACM (2019) 25-34

[3] H., Lopez-Fernandez, M.R. Jato, D.G. Pena, F. Aparicio, D Gachet, M. Buenaga and F.F. Riverola, BioAnnote: a software platform for annotating biomedical documents with application in medical learning environments. Computer Methods and Programs in Biomed, 111(1)(2013) 139–147.

[4] M. Perez-Perez, Glez-Pena, F. Fdez-Riverola, and A. Lourenco, Marky: a tool supporting annotation consistency in multi-user and iterative document annotation projects'. Comput. Methods Prog. Biomed., 118(2015) 242–251.

[5] R. Islamaj, D. Kwon, S. Kim, and Z. Lu, TeamTat: a collaborative text annotation tool. Nucleic Acids Research, 48(2020) 5-11.

[6] K. Bontcheva, H. Cunningham, I. Roberts, A. Roberts, V. Tablan, N. Aswani, and G. Gorrell. GATE Teamware: a web-based, collaborative text annotation framework. Language Resource Evaluation, 47(2013) 1007–1029.

[7] J. K. Kummerfeld, SLATE a super-lightweight annotation tool for experts, in Proc. of 57th Annual Meeting of Association for Computational Linguistics: System Demonstration, ACL,(2019) 7-12.

[8] F. Yin, Q.-F. Wang and C.-L. Liu. A Tool for ground-truthing text lines and characters in off-line handwritten Chinese documents, in Proceedings of 10th ICDAR, (2009) 951-955.

[9] C. L. Liu, Yin, D. Wang and Q. Wang. CASIA online and offline Chinese handwriting databases, in Proceedings of 12th ICDAR, (2011) 31-47.

[10] R. I. Elanwar, M. Rashwan, and S. Mashali, A Semi-automatic annotation tool for Arabic Online Hanwritten Text, Intl. Journal on Islamic Applications in Computer Science and Technology, 1(1)(2013) 19-31.

[11] I. Thompson., About World Languages. [Online]. Available http://aboutworldlanguages.com/javanese. (2016).

[12] I. Partogi, Origin of our national language, in The Jakarta Post. Depok: Jakarta Post, 20(2017).

[13] L.D. Krisnawati, and A.W. Mahastama, O Uso da Wikipedia como fonte de suporte para pesquisas em idiomas com recursos digitais insuficientes, Prisma.com, 40 34-44.

[14] SIL International., Ethnologue: Languages of the world, 22nd ed. (2019).[Online]. Available: http://www. ethnologue.com.,

[15] J. R. Buelow, T. Barry and L.E. Rich., Supporting Learning Engagement with online Students, Online Learning Journal, 22(4)(2018) 313-340.

[16] M. T. Chandio, S.M. Pandhiani and R. Iqbal., Bloom's Taxonomy: Improving Assessment and Teaching-Learning Process., Journal of Education and Educational Development, 3(2)(2016) 203-220.