Transliteration Balinese Latin Text Becomes Aksara Bali Using Rule Base And Levenshtein Distance Approach

Research · June 2016

DOI: 10.13140/RG.2.2.19430.65602

CITATION
1

READS
577

3 authors:

Made Sudarma
Udayana University
204 PUBLICATIONS **842** CITATIONS
SEE PROFILE

I Nyoman Satya Kumara
Udayana University
162 PUBLICATIONS **849** CITATIONS
SEE PROFILE

I Putu Agus Eka Darma Udayana
22 PUBLICATIONS **78** CITATIONS
SEE PROFILE

Indonesian Journal of Electrical Engineering and Computer Science
Vol. 2, No. 3, June 2016, pp. 401 ~ 408
DOI: 10.11591/ijeecs.v2.i2.pages

Transliteration Balinese Latin Text Becomes Aksara Bali Using Rule Base And Levenshtein Distance Approach

Made Sudarma[1], I Nyoman Sayta Kumara[2] , I Putu Agus Eka Darma Udayana[3]
[1,2]Department of Electrical and Computer Engineering, Faculty of Engineering, Udayana University
Jimbaran Campus, Bali-Indonesia, Ph/Fax: +62361703315
[3]Magister Program of Electrical and Computer Engineering, Udayana University Graduate Program
Jl. PB Sudirman Denpasar 80232, Bali-Indonesia
Corresponding author, e-mail: imasudarma@gmail.com[1], satya.kumara@ee.unud.ac.id[2],
agus.ekadarma@gmail.com[3]

Abstract

In the ancient Balinese Civillization, lontar is a way of Balinese people to preserving information. Lontar as a medium of writing, still taught today as a way to preserving Balinese heritage among Balinese youngster, but Lontar is struggling to survive in modern life of Balinese people ever since. One of the problem is, mainly that the lontar are not well maintained lontar also a medium of writing also found to be worn out soon so its not a good medium of storing information, lontar itself it doesn't have good endurance properties against times and natural weather. Author propose to digitize the experience of writing lontar with the use of software. The digitalization process of each letter in Balinese alphabet can be done by rewriting the script of that lontar in Balinese alphabet by using translation script powered by Rule Base and assisted with Levenshtein Distance Approach. The touch of Digital technology will preserving the lontar into a digital format thus making the problem of worn out lontar becoming obsolete making the information that written in this digital lontar protected for a long time.

Keywords*: Translitation, rule base, levenshtein distance, aksara Bali*

1. Introduction

Ancient balinese people have their own linguistic and lettering system called *aksara* Bali. Literally mean Balinese alphabet, *aksara* Bali was used since long time ago as a basis of Balinese civilization and litelature [1]. If we divide *aksara* bali structure by its phonetics *aksara* Bali letter can be devided into several ditict groups called warga *aksara*. This division method is known as Panini's principle. A way of phonetics division that are well known in ancient India. According to *Panini* principle a balinese letter system would be divided into 5 big division that consist of : *warga kanthya, talawya, murdhanya, dantya* and *osthya* [2].

Balinese lettering system or *aksara* Bali, find itself struggling in modern era. When the uses of it is replaced by latin lettering system, endorsed by dutch colonication of Indonesia archipelago. Balinese lettering system is found only in Bali traditional parypus like called *Lontar. Lontar* is one of traditional writing medium that has been used by the ancestor of Balinese people which their uses has been documented as old as long time ago. *Lontar* often described as a ancient script by today modern balinese people often contains an old relogious scripture as well as a legal and custom law, or ancient epic story of ancient balinese civilization [3]. Thats one of many reasons why people of Bali is trying their best to preserving the lontar, even so far to copy every *lontar* in existence .But this preservation initiative often faced a lot of challenge on their way. First,is the nature of *Lontar* leaves itself that often worn out overtimes, that makes the documentation effort kinda problematic. Two is the process of copying it is a very complicated and time consuming task, as it requires an expert knowledge and a lot of manual work And third is limited financing by the government to fund the initiative to take care all these documents.

This inplementationn of balinese script writing with digital approach is in line with a local goverment initiative. With the regional act that appointed on S.K. nomor 179 tahun 1995 and circulation letter no 01/1995 that dictate the uses of *aksara* Bali as a legal form of writing On

Received January 1, 2016; Revised April 24, 2016; Accepted May 10, 2016

the other side there is another efforts from the government in preserve the Balinese alphabet in their native form of l*ontar*. But this initiative sure comes with its own challenge as.

Preserving the Balinese culture in form of *lontar* only is not enough as we think the way of *aksara* Bali written itself should become modern. Specially should get some kind of computerization involved citing the initiative that has been done before in the papers explain about translation had been done by using Syllabification Approach method [5]. And also the other one that address some grammatical problem in the paper entitled Rule Base, Decision Treedan Hybrid [6][7][8]. And this one with the similar approach that takes emphasis of structural problem inherent in the language [9]. This paper combine the problem faced with 3 previou reseacher, point an improvement of the rule base aspect and also giving the addition of spellchecker that powered with Levenshtein Distance method. This combined approach is used to decrease the level of gramatical mistake that often faced during translation.

2. Fundamental Theory

2.1 Balinese Alphabet (Aksara Bali)

Balinese alphabet is one of the heritage of Balinese people, the uses of balinese alphabet is as old as balinese society itself [10]. From the historical point of view, the uses of balinese alpabet is having a close ties grammatically with the alphabet found in India. Expert theories the india alphabet system fused into balinese culture when Hindu and Buddha came to Indonesia [11]. *Aksara* (Alphabet) phonetically means a visual symbol system which used to store a data on certain medium and has a role to aid people to express thought in Balinese language back in the ancient times [12]. Based on its phonetics *aksara Bali* (Balinese alphabet) is divided into some groups which is called *warga akasara* [2]. There is a local convention called *Pesamuhan Agung* which is done in 1963 that have decided a set of characted that formally used to be *aksara* Bali complement of latin alphabet in general, thus becoming the tools to represent a latin text [13]. *Aksara suara* or a set of character that having a vocal similarity with a latin text [14]. In *warga aksara* there is a special set of consonant only character also called, *Aksara Wianjana*. Table 2 is the classification of *Aksara Wianjana* based on *Warga Aksara*.

Table 1. Aksara Suara

Number	Balinese script	Latin script
		a
		ê
		i
		u
		e
		o

Table 2. Aksara Wianjana

Number	Balinese script	Latin script	Number	Balinese script	Latin script
		h / a			l
		n			m
		c			g
		r			b
		k			ng
		d			p
		t			j
		s			y
		w			ny

2.2 Lontar

A old ways of writting in Bali, insted of using a stone tablet or other hard medium. Old Balinese people using a *lontar* instead. *Lontar* (Java: *ron tal, "daun tal"(*tal leaf*))* is siwalan leaf or tal (*Borassus flabellifer* atau *palmyra*) that had been dried. *Lontar* tree (*Borassus flabellifer*) is a kind of *palma* (palm) which grow in South East and South of Asia. In Indonesia Archipelago, the relic of *lontar* usually been found in some area such as in Bali, Java, Lombok and Sulawesi. That *Lontar* is used as place for writing before paper [15].

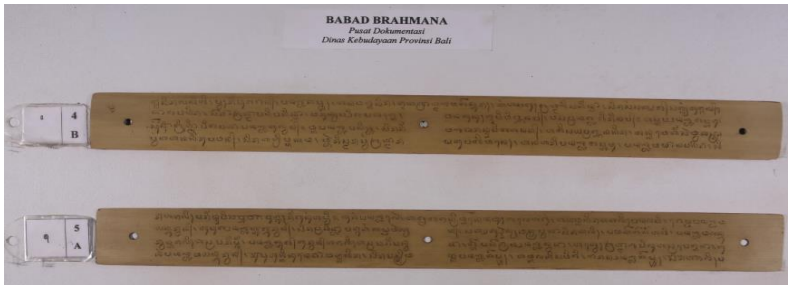

Figure 1. Capture Lontar Babad Brahmana

2.3 Transliteration

Transliteration literally meaning a progress of changing words in one language into the other language and still have the same structure [9]. Transliteration can be as a words conversion progress which is written in one language into another language too by keeping the pronunciation. In this case, you will hear about transliteration and translation. A lot in this paper Transliteration is the more advanced form of translation as translation will simply just meaning interpret from the text or communicate the same message from original language.

2.4 Tokenizing

Tokenizing is a process of dividing a text to become some kind of "token" token means the structure of characters which represent the certain word and a product of tokenizer [16]. Tokenizer divide a string into some tokens based on the certain characters. In tokenizing progress, the characters usually divided as guidance for doing the segmentation sentence become token such as space, tab, new row (white space character). This is the example of tokenizing in the sentence "*ida sampun ngerayunan* "

Input : *ida sampun ngerayunan*

Output : *[ida] [sampun] [ngerayunan]*

2.5 Spell Checker

Spell checking or spell correction is some kind of additional module that usually found in text processing application that query used as an input will be corrected the spelling/writing depending on the dictionary used. Algorithmm used with the spellcheker that included in ths paper is using a Levenshtein Distance*, that is basis are* the sets of calculation of edit distance from a certain matrix which is later used to count the total between two string [17].

2.6 Rule Base

Rule Base method is a algorithm that we used to build a language machine in this paper and govern the rules in doing the transliteration progress [9]. Other than rules, involved a dictionary data that create sets or corresponding word set for every word in the two languages. So, every word is translated one by one, and then arranged again based on the basic language rule. The system depends on the linguistic knowledge provided before in the database. This approach had. Their own benefits of being able to analyze the bevahiour of the program in every input scenario till syntaxes level and semantic. The weakness is the need of domain expert in order to write the rule based machine, thus limiting the machine translation quality to be just as good as what expert provided.

3. Research Method

3.1 Design And Implementation

The working scheme of the system is divided into 3 big subroutine there are Latin text input progress, the saving progress of the tokenization data, spell checker and finally the transliteration subroutine that would be showing the result of translation in *aksara* Bali. Picture 2 is visualize the inner working which happen in the program transliteration Latin text become *aksara* Bali in the form of simple block diagram.

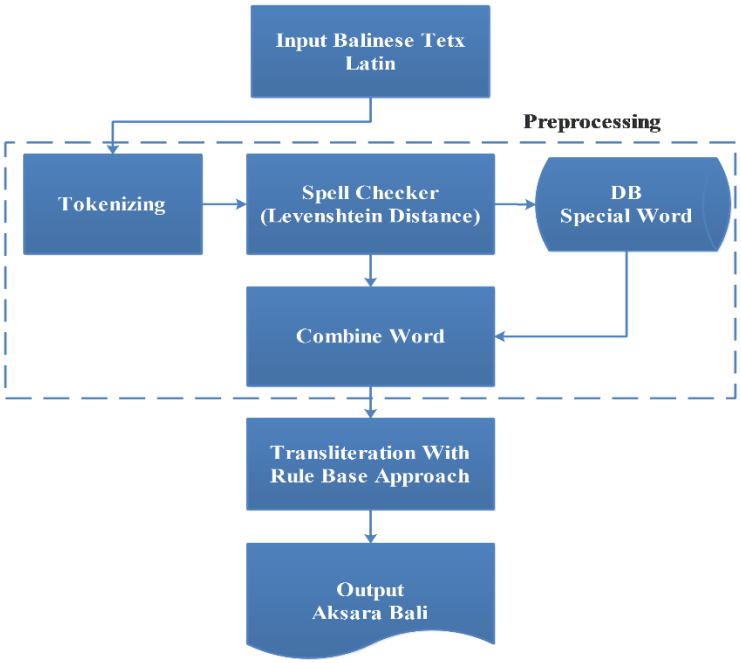

Figure 2. Block Diagram of System

3.2 Tokenizing Process

Tokenizing progress which is fall into the preprocessing subroutine in the transliteration Latin text becomes *aksara* Bali. In this subroutine Latin text which is inputted will be read by system. When all texts have have been read by the system then that texts will be divided into some groups based on certain delimiter, in general the space will be deleted. And every words which have be processed earlier to delete the space between word will be given its own unique sign. That sign will be used to arrange word and the result will be saved to be later used later in the next progress

3.3 Spell Checker

Spell Checker subroutine is there in the system to tackle the need of checking multiple meaning and homonym across balinese dictionary Spellchecker subroutine is done by let the program match every word in data base which is saved by the system. On the first step will be done the initialization the total of words in the text which is inputted by the system. Those words will be inputted into Levenshtein Distance algorithm to do counting the space between source words and destination words in data base. The word which doesn't have space will be showed as suggestion word which become the purpose of the user and will be included with the meaning of that suggestion word by using direct mapping. The suggestion word can be meant as word that has more than one meaning or homonym, so, when it comes in to transliteration progress, there will not happen mistake in the*Aksara* Bali output. This is because of the writing of *aksara* Bali, the word which notabene has same writing and may different result in Balinese language. Talking about spell checker, this progress will continue well from the checking word or checking the space of word by Levenshtein Distance method till all words which has inputted complete.

3.4 Levenshtein Distance

In this subroutine, there is a special subroutine that implement Levenshtein Distance algorithm, which is looking for the equivalent matrix value in every word that has been inputed compared with word that is inside the transliteration database system. Every word that come into this subroutine will be faced with some preprocessing as stated in the software design phase naming tokenization and other stuff. The one of the said preprocessing is Levenshtein Distance algorithm that are implemented to change the character into the correct grammar to make that thing to be done, there is a few operation in place, first increasing character and second deleting character. In changing character operation, a character will be subtituted into

other character. In increasing character operation the string will be added another character into string. In deletion character operation is done by deleting character in the string. Based on that condition, the prechecking iniative are put into place naming if the last word from the first text more or less from the second text, the process would be skipped if the string on the first text and second text has difference, so there will be the checking and changing progress. Said above those progress will always done inside a loop until all string in second text which is equivalent come into Levenshtein Distance algorithm sucess flag.

3.5 Rule Base Approach

The core algorithm in transliteration subroutine that running on this system is the inplementation of so called rule base approach. Every character which is in the text present in the user input bar will be checked by the set of rule which later decided how the program willchange the latin text to *qksara* Bali. One character at the time when the first character come into rule base algorithm, that character will be compared to the set of rule hardwritten in the code . When that character is not in the first set of rule, that character will be hopped into other rule in the system. When the latin character has come into one of the rule which has prepared by the system and the flag comes out true, then that character will be saved in working memory, and it will be changed into corresponding character.

4. Result And Analysis

An act of evaluating the transliteration system performance, which measures the result of the system that has been built. The results of every tested which has been done such as.

4.1 Testing of Transliteration

The testing is about how the application which had been developed doing transliteration process by using Rule Base method. The menu consist of execution process, importing, saving the result, redo and exit. The general uses of the program would be presented in figure 3. In figure 3 is seen that the result of translation latin text in *lontar* has been completed to do transliteration or rewrite in *aksara* Bali form. To save the result of *aksara* Bali if you click the save menu.

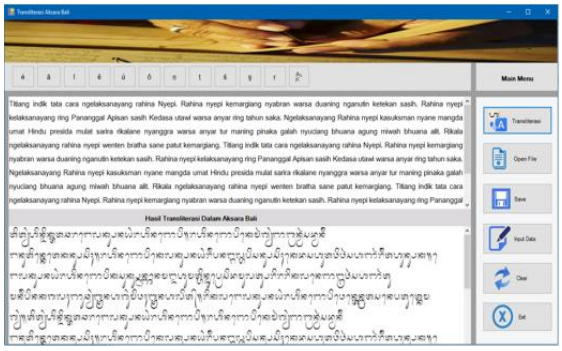

Figure 3. The Result of Translation

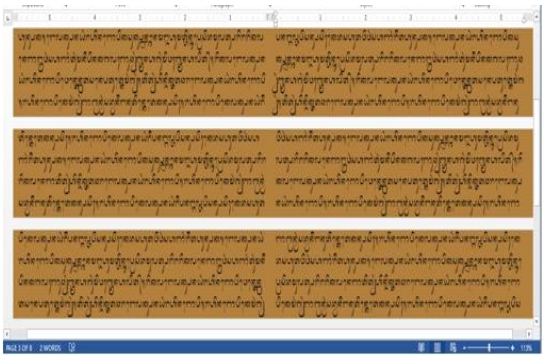

Figure 4. The Result of Rewrite Lontar in Document Form

Figure 4 is the result of transliteration translation document that have a standard word document but with a format of traditional *lontar evalueate* the accuracy of transliteration application in using rule base method. In this testing is using 20 pieces of *lontar* documents to match with the result of transliteration documents. The result of that testing is the accuracy value which is produced from the result of transliteration by using rule base method and supported with levenshtein distance method which result can be seen in section 4.2.

4.2 Accuracy Testing

Accuracy testing is done to the system as a way to test the accuracy of the inplemented algothirm and solution. In the accuracy testing is used 20 sheet of manual translation *lontar* in balinese latin text used as input of application. This manual transliteration document in balinese latin text of *lontar* is obtained from Pusat Dokumentasi Dinas Kebudayaan Provinsi Bali. The system would be tasked of transliterating every sheet on manual transliteration document in balinese latin text to *aksara* Bali. The result of this Balinese script transliteration will be pressed

with the original *lontar* script and also the transliteration result will be tested by expert user in *aksara* Bali. The result would be populated below in the result table 3.

Table 3. The Result of Accuracy Testing

No	Lontar Code	Number of Character	Right Tertransliteration	Wrong Tertransliteration	Accuracy (%)
1	1B	498	487	11	97.79%
2	2A	491	488	3	99.39%
3	2B	483	481	2	99.59%
4	3A	545	545	0	100.00%
5	3B	503	501	2	99.60%
6	4A	549	548	1	99.82%
7	4B	509	508	1	99.80%
8	5A	546	539	7	99.72%
9	5B	524	521	3	99.43%
10	6A	540	540	0	100.00%
11	6B	532	529	3	99.44%
12	7A	546	541	5	99.08%
13	7B	600	589	11	98.17%
14	8A	565	559	6	98.94%
15	8B	560	553	7	98.75%
16	9A	565	551	14	97.52%
17	9B	546	539	7	98.72%
18	10A	540	540	0	100.00%
19	10B	600	589	11	98.17%
20	11A	565	559	6	98.94%
			Average		**99.09%**

The figure below show the inspection given by the domain expert of balinese language. The result of the program is compared with the manual translation given by the group of expert as control. The result of the test would be accumulated and given the average below. The average translation yield 99.09 % result.

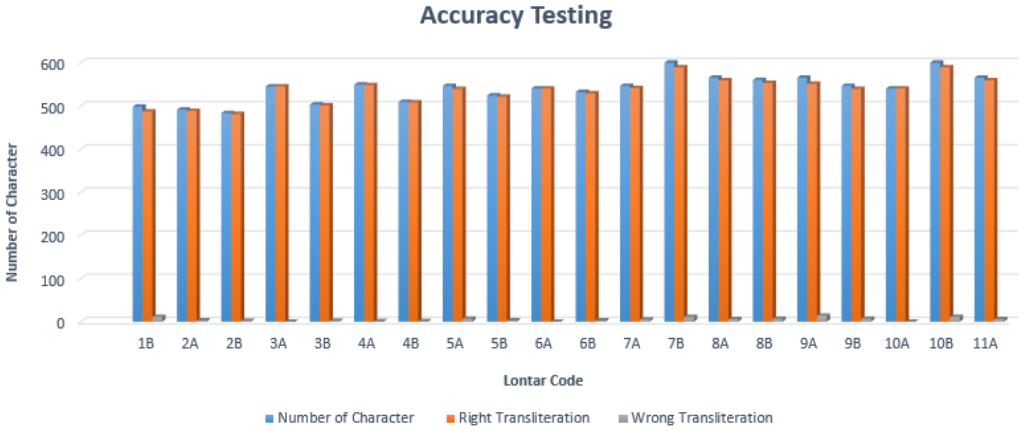

Figure 5. Block Diagram of Akurasi Testing

The figure 5 populated the transliteration plot and added 3 data point the first one is total character the wrong / mistranslated result and the correct translation. According to accuracy testing the scripture code 9A has the most wrong charater to be translated. From a group of 565 character it only wrong 14 out of 565 and best case scenario would be found on scripture code 3A, 6A and 10A with no mistranslated character at all.

In general the error that are found in the system can be divided into 3 major reasons the first one is there is some kind of inconsistency found in the scripture. Experrt says that the linguistic maturity of the balinese people is a major issue that caused this. The second one is there is found a new character in the new scripture a character set that not yet defined by the system database thus cant be translated correctly. And the thrid one is a human error that are found in an achive data that given by balinese council of balinese text preservation comitte. The human itself is caused with balinese grammar always evolves over time.

4.3 Usability Testing

The usability testing is given to meassure the quality of the system in the term of user friendlyness. To meassure this we uses eight people of blind tester from balinese commite of language preservation. Every tester would be given the set of task that has to been completed after completing the task every tester should put up some quistionary with a set of predefined question that are made based on Nielsen's Approach, which the 5 aspect explained is learnability, efficiency, memorability, errors dan satisfaction. The result of the test would be given in table 4.

Table 4. The Result of Usability Testing

No	Question	Average
System Aspect		
1	Is the aplication is easily identified	3.75
2	Is the application easy to use	3.64
3	Is the color scheme okay dan easy to remember	3.38
User Aspect		
4	Is the menu not confusing	3.75
5	Is there is a clear distinction between input and output form	3.88
6	Is the output application easy to understandable	3.50
7	Is the execution button easy to find	3.88
8	Is the icon is easy to understand	3.63
Interaction Aspect		
9	Is easy to input latin text on this aplication	3.63
10	Is the output format given (microsoft word) helping much	3.88
11	Is the transliteration output in aksara Bali easy to find	3.50
12	Is the database given in this application easy to manage	3.25
13	Is the menu position memorable	3.75
14	Is the input and output form memorable and easy to spot	3.50
15	is the application suits to needs	4.00

Tabel 4 shown the result of the usability testing. It can be seen that all the attributes contained in usability testing have satisfaction values above 3.00 and in point 15 of the table giving the average of 4.00 and if we conclude the average would be 3.66 on all aspect which is really good given the max scale is 4.00. The result would be making the effort that have been put in place yield a very good result and for point 15 is demonstrated application developed is needed by Pusat Dokumentasi Dinas Kebudayaan Provinsi Bali.

5. Conclucion

The conclusion of the reseach on transliteration application of Latin text in Balinese language into *aksara* Bali by using Rule Base method is able to rewrite the transaltation of latin text of *lontar* into *aksara* Bali as like in the original *lontar* document with 99.09 % accuracy. Based on the result of usability testing, the average user satisfaction of the application on all aspect would be 3,66 which means the user is satisfied with the application. The next study can be done the addition word data bank in Balinese language which has special writing in *aksara* Bali or making the best method for higher result.

Acknowledgment

The appreciation of the writer gives to colleagues and all sides which had helped provide critical comments to make this research better.

References

[1] M. Sudarma, S. Ariyani, and M. Artana. Balinese Script's Character Reconstruction Using Linear Discriminant Analysis. *Indones. J. Electr. Eng. Comput. Sci.* 2016; 4(2); 479-485.
[2] I. M. Surada. Kamus Sansekerta-Indonesia. Surabaya: Paramitha. 2007.
[3] I. K. Ginarsa. The Palm (Palmyra) Palm. Edisi Dua Bahasa. Denpasar. 1976.
[4] I. N. Mahendra, M. B. Suasta, I. W. O. Granoka, and I. W. Japa. Pembinaan Bahasa Aksara Dan Sastra Bali Pedoman Penulisan Papan Nama Dengan Aksara Bali. Bali: Dinas Kebudayaan Propinsi Daerah Tingkat Bali. 1996.
[5] H. Joshi, A. Bhatt, and H. Patel. Transliterated Search using Syllabification Approach. *Forum for Information Retrieval Evaluation*. 2013.
[6] D. Bhalla, N. Joshi, and I. Mathur. Rule Based Transliteration Scheme for English to Punjabi. *Int. J. Nat. Lang. Comput.* 2013; 2(2); 67–73.
[7] V. Kaur, A. kaur Sarao, and J. Singh. Hybrid Approach for Hindi to English Transliteration System for Proper Nouns. *IJCSIT*. 2014; 5(5); 6361-6366.
[8] E. S. Kaur, M. R. Kaur, and E. N. Bhalla. Architecture of the Transliteration System from ENGLISH to PUNJABI using Hybrid approach. *International Journal of Computer Science and Information Technology & Security (IJCSITS)*. 2012; 2(2); 482-485.
[9] D. Kaur and E. R. Kaur. English To Punjabi Script Converter System For Proper Nouns Using Hybrid Approach. *IJARCSSE*. 2014; 4(7); 551–554.
[10] I. Darma, I. K. Putra, and M. Sudarma. Ekstraksi Fitur Aksara Bali Menggunakan Metode Zoning. *Maj. Ilm. Teknol. Elektro*. 2015; 14(2); 44-49.
[11] M. Sudarma and I. W. A. Surya. The Identification of Balinese Scripts' Characters based on Semantic Feature and K Nearest Neighbor. *Int. J. Comput. Appl.* 2014; 91(4); 14-18.
[12] http://ensiklo.com/2014/09/istilah-aksara-berasal-dari-bahasa-sanskerta-yang-berarti-tidak-musnah, cited: September 12, 2014.
[13] M. Sudarma and N. P. Sutramiani. The Thinning Zhang-Suen Application Method in the Image of Balinese Scripts on the Papyrus. *Int. J. Comput. Appl.* 2014; 91(1); 9-13.
[14] N. P. Sutramiani, Ik. G. Darmaputra, and M. Sudarma. Local Adaptive Thresholding Pada Preprocessing Citra Lontar Aksara Bali. *Maj. Ilm. Teknol. Elektro J. Electr. Technol.* 2015; 14(1); 27-30.
[15] M. Sudarma. Identifying of the Space Color CIELab for the Balinese Papyrus Characters. *Int. J. Soft Comput.* 2015; 11(2). 64-69.
[16] A. Indranandita, B. Susanto, and A. Rahmat. Sistem Klasifikasi dan Pencarian Jurnal Dengan Menggunakan Metode Naive Bayes dan Vector Space Model. *J. Inform.* 2011; 4(2).
[17] N. M. M. Adriyani. Implementasi Algoritma Levenshtein Distance Dan Metode Empiris Untuk Menampilkan Saran Perbaikan Kesalahan Pengetikan Dokumen Berbahasa Indonesia. *JELIKU-J. Elektron. Ilmu Komput. Univ. Udayana*. 2012; 1(1).