Journal of Theoretical and Applied Information Technology
15th October 2017. Vol.95. No 19

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A HYBRID METHOD OF RULE-BASED AND STRING MATCHING STEMMER FOR JAVANESE LANGUAGE

[1]FATKHUL AMIN, [2]WIWIEN HADIKURNIAWATI, [3]SETYAWAN WIBISONO,
[4]HERNY FEBRUARIYANTI, [5]JATI SASONGKO WIBOWO
[1,2,3,4,5]Faculty of Information Technology, Universitas Stikubank, Semarang, Indonesia

E-mail: [1]fatkhulamin@edu.unisbank.ac.id, [2]wiwien@edu.unisbank.ac.id, [3]setyawan@edu.unisbank.ac.id,
[4]hernyfeb@edu.unisbank.ac.id, [5]jatisw@edu.unisbank.ac.id

ABSTRACT

Language is rich in morphological variations but poor in linguistic computational resources. Ngoko Javanese language is a morphologically rich language that has a different variant form of words. This paper describes an algorithm by which a stem for Ngoko Javanese language. Ngoko Javanese language stemmer is efficiently used in information retrieval. Through this algorithm, we can get a root from its actual word. We use a hybrid rule-based and string matching algorithm. Special rules are created to remove the prefixes and suffixes of the Ngoko Javanese terms. The algorithm has been tested on hundreds of Ngoko Javanese words. Results reveal that the accuracy reaches to about 67%.
Keywords: *Javanese Language, Stemming, String Matching, Rule Based Algorithm*

1. INTRODUCTION

Recently, improved information retrieval has become necessary because of the huge amount of information people have. The Information retrieval is the science of retrieving a subset of documents that satisfy the user's need from a collection of documents [1].

The main objective of information retrieval is to analyze and treat documents to extract some measures and relevant data allowing users for their information need. One of the first steps in the information retrieval process is stemming [2]. Stemming is a branch of morphology. Morphology means a study of word structure. Stemming is a process of extracting root word from its actual word and separate the affixes [3][4]. A stem is a combination of a root and derivational morphemes to which an affix or some affixes can be added [5]. The root word is determined by removing the affixes that carry grammatical or lexical information about the word. In both cases, these affixes don't modify the concept of the word related to as the semantic informality that has been proven in the literature. Stemming plays a vital role in the field of Natural Language Processing (NLP). It is important and useful in the spelling check, translation machine and especially information retrieval [3].

The main purpose of stemmer is to reduce the inflected words into its root form or stem by removing the affixes. By stemming, words in inflectional and derivational forms can be mapped into the same concept [6][7]. The confusion of the words sharing the same stem leads to a reduction of the dictionary to be taken into account in the process, as the whole vocabulary contained in the input unprocessed collection of documents can be reduced to a set of topics or stems. This leads to a reduction of the space needed to store the structures used by an information retrieval system (like the index of terms-documents) and then also lightens the computational load of the system [5]. In computational linguistics, a stem is the part of the word that never changes even when morphologically inflected. Stemmer research is done on the scope of text retrieval. Stemmer in this study aims to improve retrieval sensitivity by increasing the ability to find words in relevant documents. The main importance of stemmer is in information retrieval application where a stemmed query proportionally leads to increase in recall for the given application.

Javanese language stemming has a complex problem because of the many affixes to its root word. The affixes are prefixes, suffixes and infixes. In addition there are different types of these affixes, so to get the root can not only be done by a rule based method stemming. The addition of string matching method is useful when the root is not found by using the rule based method. The similarity of the word becomes the key to the string matching method. In this stemming the word base is assumed to be the word

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

that has the highest similarity to the word in the dictionary. The addition of string matching method will reduce the result of the wrong stem. In addition, it can reduce over stemming and under stemming.

This paper is organized as follows: In Section 2, we give an overview of background and related works. In Section 3, we present our procedure or algorithm used to find the stem of words used in Javanese Indonesian language. In Section 4, we discuss the testing and evaluation results. In Section 5, we talk about the conclusion.

2. BACKGROUND AND RELATED WORKS

People's attitude toward a language is different. It is known as a variety of language. The official language in Java Island is the Indonesian language; meanwhile, the Javanese itself is used as a communal language in sub-urban area of Java. There are some differences among dialects of Javanese Language in general because of many factors. The factors that influence the dialect differences are: 1. Geographical Aspect, 2. Age and Gender, 3. Education Level, and 4. Social Class. The different social class will produce different utterances of dialect. In Javanese language, it has a hierarchy of stratification which is divided into three categories: 1. Ngoko Javanese Language, 2. Kromo Javanese Language, 3. Kromo Language. Ngoko Javanese Language is used by people who are already close to each other, to people in the same age or to those who are lower in social status [8]. It is used as an object of stemmer on this research.

The Javanese language is a morphologically rich language that it has a different variant form of words. Javanese dialects add a set variation of dimensions in some fields like natural language processing and information retrieval. There is no stemming algorithm that handles Ngoko Javanese dialect. Only a few algorithms are available that handle only a singular Ngoko Javanese dialect. The objective of this paper is to describe an algorithm by which a stem for Javanese dialect can be defined.

Many of the stemmers are aimed at and evaluated with the English language. Some researchers have modified existing approaches or proposed a new technique to handle other languages. Researchers have developed stemmer in addition to English, for example Indonesian stemmer [9], Finnish stemmer [10], Hungarian and Czech stemmers [11], Portuguese stemmer [12], Turkish stemmer [13][14], German stemmer [15], French stemmer [16], Greek stemmer [17], Arabic stemmer [18][19][20], and Urdu stemmer [3].

Each stemmer can use different methods. This research was developed based on 5 key papers summarized in Table 1.

Tabel 1: Key Papers for Research

No	Author	Method	Reseach Focus	Language and accuracy
1	Ali et.al (2016)	Rule based	1. Remove prefix, infix and postfix 2. Competent to generate the stem of borrowed words and compound words	Urdu, 90 %
2	Pandey et.al (2016)	Rule based, wordnet based	1.Suffix removal. 2.Inflection removal	Marathi
3	Meitei et.al (2015)	Hybrid method (Brute force algorithm and Suffix stripping algorithm)	Reduce over stemming and under stemming	Manipuri, 86 %
4	Gupta et.al (2015)	Rule based, dictionary based	1.Developed inflectional and derivational rule based stemmer, 2.Reduce over stemming and under stemming	Urdu, 89 %
5	Abuata and Al-Omari (2015)	Rule based, dictionary based	1. Remove suffixes and prefixes. 2.This algorithm applies	Arabic Gulf Dialect

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

			word size rules (at least 3 letters) and the relation ship between adjacent letters.	

The proposed method of this research using hybrid method to improve stemming result. The results to be achieved is to reduce over and under stemming and improve precision and accuracy. The hybrid method is rule based and string matching.

3. STEMMING

The way to extract the root of Javanese word is to identify what the excessive letters are in the word. There are two points to be considered while using a stemmer: 1. Morphological forms of a word are assumed to have the same base meaning and should be mapped to the same stem, 2. Words that don't have the same meaning should be kept separate.

Stemmer is expected to be able to reduce the dimension of data; so it will improve the performance of the categorization process. The diminishing dimension of the data causes the smaller rules to determine a categorized document. It can improve the categorization result. Reduction of data dimensions using stemming is caused by words that have the same root word classified into one attribute. It causes the dimension of data can be reduced.

The stemming algorithm for one language is different from another language. The process of stemming for Ngoko Javanese Language is more complicated because there are variations of affixes that must be removed to get the root word. The use of appropriate stemming algorithms affects the Information Retrieval (IR) performance. The removing process of prefixes and suffixes in stemming must be adjusted to the rules applicable in the language.

3.1 Ngoko Javanese Language Stemmer Model

Language undergoes several processes, namely affixation, repetition and compounding. affixation can be done by prefixation, infixation and suffixation.

Ngoko Javanese Language stemmer is created using the morphological approach of Ngoko Javanese Language. Affix Removal method is used because it is flexible as a stemmer of various languages with its characteristics. It is more emphasis on the morphological structure of a language. This method will remove prefix, suffix and infix of a term and convert them to root.

The stemming is used to remove the prefix, suffix and infix contained on the word and then processed to become a root word. Ngoko Javanese words consist of prefixes, infixes, suffixes, repeated forms, and confixes. Ngoko Javanese stemming has some very specific problems with the language. One of those problem is the different type of affixes. Prefixes may change depending on the first letter of the root word. For example, prefix 'ng' changes into 'k' when the first letter of the root word is 'g', e.g. 'ngetok' (root: ketok). Another rule in which prefix can be changed into 'ng' is when the first letter of the root is 'k', for example 'ngethok' (root:kethok). 'Ketok' and 'kethok' have different meanings. In English, 'ketok' means visible and 'kethok' means cut. If there is more than one affix that is attached to a word then the sequence of steps to remove the affixes becomes very important. If not paying attention to the sequence of removal steps then the root word is not found in the dictionary. Limitations of this stemmer are 1. words that have no affixes will not be stemmed process. 2. The same affix is never repeated. 3. The character will be returned after the prefixing process. In this stemming process, the first step is to check affixation. Affix that becomes the input stemming first checked on the dictionary data base. If the word is found in the dictionary then the word has become the root word. If the word is not contained in the dictionary then the word will be cutting process. If the base word has not been found after going through the rule based stemmer then the next step through string matching. Matching of words is done to find the highest similarity with the word in the dictionary. If all the stages have been passed without generating the root word then a word that has a high similarity is assumed to be the root word. The block diagram of the proposed stemming process can be seen in Figure 1.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

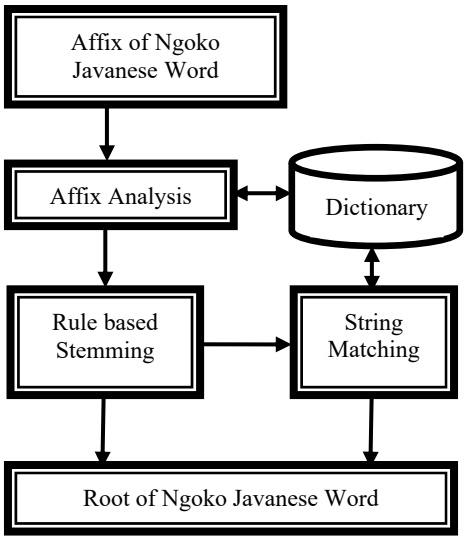

Figure 1. Diagram of Proposed Stemming Process

Ngoko Javanese Language stemmer is adopted from the Porter stemmer algorithm. It uses rule-based analysis to find the root of a word. The stemming algorithm of Ngoko Javanese language uses a rule-based stemming algorithm which has several steps as follows:

1. Eliminate prefixes (prefix-4, prefix-3, prefix-2, and prefix-1).
2. If a rule has been done,the next steps are:
   a. Eliminate the suffixes (suffix-3, suffix-2, and suffix-1)
   b. If a rule has been done, remove the prefix and infix, otherwise the stemming process is completed.
3. If no rule is done, run the following steps:
   a. Eliminate prefixes,
   b. Eliminate suffixes,
   c. Eliminate infixes,
   d. The stemming process is completed.

The stemming process is designed to replace omitted prefix with the corresponding letter. The process of removing prefixes, suffixes and replace infixes are done in one stage. Figure 2 shows the flowchart of stemming.

The design of Ngoko Javanese language stemmer user interface is displayed word-by-word. Each word will do through word separation process with prefix and suffix. The stemmer design can be seen in Figure 3.

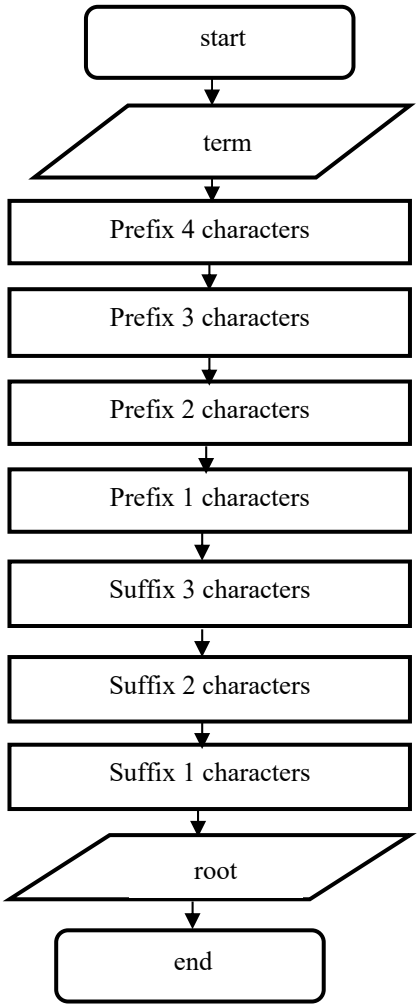

Figure 2. Flowchart of Stemming Process

Title	**:**
Term (Affix)	**:**
Frequency	**:**
Prefix	**:**
Root	**:**
Suffix	**:**

Figure 3. User Interface Performance of Stemmer

3.2 Eliminating Prefixes

The process of removing the prefix aims to get the word by separating the prefix with its root word. In Ngoko Javanese language, the number and the type of prefixes can be seen in table 2.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: Types of Prefixes

Types of Prefixes	Prefix	Examples
Prefix 4 characters	kuma-	kuma + wani = kumawani
	kapi-	kapi + lare = kapilare
Prefix 3 characters	dak-	dak + gawa = dakgawa
	kok-	kok + jupuk = kokjupuk
	pan-	pan + gayuh = panggayuh
	pra-	pra + lambang = pralambang
	tar-	tar + buka = tarbuka
	tak-	tak + antem = takantem
	tok-	tok + simpen = toksimpen
Prefix 2 characters	di-	di + pendhem = dipendhem
	ka-	ka + boyong = kaboyong
	ke-	ke + thutuk = kethutuk
	ma-	ma + gawe = magawe
	pa-	pa + warto = pawarto
	pi-	pi + wulang = piwulang
	sa-	sa + wiji = sawiji
Prefix 1 character	a-	a + gawe = agawe

3.3 Eliminating Suffixes

The process of eliminating the suffix aims to get the root word by separating the suffix with its root word. Some suffixes in Javanese with examples of their usage are shown in table 3.

Table 3: Types of Suffixes

Type of Suffixes	*Suffix*	*Example*
Suffix 3 character	-ana	silih + ana = silihana
	-ane	jupuk + ane = jupukane
	-ake	buka + ake = bukaake
Suffix 2 character	-an	tulis + an = tulisan
	-na	gambar + na = gambarna
	-ne	rayi + ne = rayine
	-ku	omah + ku = omahku
	-ke	turu + ke = turokke
Suffix 1 character	-a	tuku + a = tukua
	-e	dhuwit + e = dhuwite
	-i	pakan + i = pakani

4. TESTING AND EVALUATION

The Ngoko Javanese language stemmer testing needs a basic word listed in the dictionary to get a good stemming result. The dictionary is needed to check the truth of the basic word resulted from the stemming process. The prefix 'a-' and 'dak-' cannot be eliminated in the stemming process of the Ngoko Javanese language stemmer test. The test result for other prefixes generates the correct root word. There are 16 prefix types successfully removed in this stemming process. The result of stemming test for the purpose of removing the suffix has been done. There are eight types of suffixes that can be properly processed. They are '-i', '-ake', '-ane', '-ana', '-na', '-ku', '-mu', and '-en'. In addition, there are three types of suffixes that fail in the stemming process. They are '-e', '-ke' and '-a'.

The test data used in this research are sentences in Javanese language magazine namely "PenjebarSemangat". The tested data consisted of 1511 raw terms that included affixes and single words.

Testing was done by entering Ngoko Javanese language articles. They consisted of thousands of words. The first process was a tokenizing process. It separated word by word. The next process was a filtering process that removed the unused characters in the articles. Unused characters were punctuations, numbers or other special characters. After being a clean word through a filtering process, these words were ready for stemming process. The stemming process was based on prefix, suffix and infix rules. The adjective may contain allomorph. Allomorph occurs on words that have an additional character 'n' or 'm'. Checking was done by matching the existing allomorph on the adjective word as input on the stemming process and then proceed the character smelting process.

After testing, the stemming process produced the correct and wrong outputs. Some outputs produced an error on the root word. The output of the Ngoko Javanese language stemming system can be seen in Figure 3. The stemming process in this study was less successful to remove infix. The wrong word stemmer results can not eliminate infix.

We also want to know that accuracy of this system. It means how much we get an accurate result. This research uses F-Measure technique to measure the performance of the proposed model. Measurement of F-Measure is based on Precision and Recall values. Recall is the proportion of root words that can be generated in the stemming process. Precision is the proportion of the number of root words produced in the stemming process and is true [21]. The supporting tables shown in Table 4 are used to help calculate recall, precision, F-Measure and accuracy values.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4. Guidelines for Calculating Recall, Precision, F-Measure and Accuracy

	Relevant	Non Relevant
Retrieved	tp	Fp
Not Retrieved	fn	tn

Tp (true positive) is the number of root words of true value and is the result of stemming, fp (false positive) is the number of words of wrong value, fn (false negative) is the number of words affixed and no stemming and tn (true negative) is not a word affixed and not stemmed

The value of recall, precision, f-measure can be calculated by the following formula:

$$Recall = \frac{tp}{tp + fn} \quad (1)$$

$$Precision = \frac{tp}{tp + fp} \quad (2)$$

$$F - Measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (3)$$

$$Accuracy = \frac{tp + tn}{tp + fp + fn + tn} \quad (4)$$

Each document is done stemming process The evaluation of stemming system results using formulas (1), (2), (3) and (4) can be seen in table 5.

There are other ways to calculate the accuracy of the system in percentage, we also use equation 5.

$$Accuracy\ (\%) = \frac{Accurate\ stemmed}{Total\ number\ of\ given\ word} \times 100\% \quad (5)$$

From several tests that had been conducted and with the calculation using the equation 4, they obtained that the rule-based algorithm had an accuracy of 67 %. Calculating the accuracy of stemmer results using formula 5 yields an accuracy of 66.7%. The success of testing stemming algorithm cannot reach 100 % because of two reasons.

a. Similarities of word that have resemble affixes lead to over stemming and under stemming. This causes the wrong root word. When a word is under-stemmed, it becomes more difficult to identify if two words are related based on their morphology, as their obtained stems are mostly equal, but not totally the same because of a suffix that has not been deleted. The opposite case takes place when over stemming is done.
b. The existence of a noun or person name that is not on the dictionary or databases causes a stemming error.

The test used 10 documents consisting of 2631 words. These words were processed on tokenizing and produced 1511 words. There were 430 words ready as input in the stemming process. This stemmer produced 335 true words and 78 wrong words. The result of Ngoko Javanese language stemmer for 10 documents can be seen in figure 4.

5. CONCLUSION

The paper describes a new algorithm to find the roots for Ngoko Javanese Language. We use a rule-based algorithm and string matching. Special rules are created to remove the prefixes and suffixes of the Ngoko Javanese terms. String matching can minimize the wrong stem result. String matching can improve root search if affixes removal steps have not found the root word. The algorithm is built on a set of predefined rules for Ngoko Javanese Language. The algorithm has been tested on hundreds of Ngoko Javanese words. The proposed system can report an average recall of 0,343, an average precision of 0,815, an average f-Measure of 0,492 and an accuracy of 66 %. The proposed rules have demonstrated an improvement in terms of providing an appropriate representation for the words. This system also ensures valid root words as the output.

6. FUTURE WORK

Although a lot of research work has already been done in developing stemmers there still remains a lot to be done to improve recall as well as precision. Required development of Java language stemmer to improve recall precision results. In future, we must do with the help of increasing affix list and update the database.

Further hybrid system can be developed using rule based and machine learning approaches to tune the system to handle more unknown words and generate more rules for the process of stemming.

7. ACKNOWLEDGEMENT

We are very thankful to Balai Bahasa Yogyakarta for the guidance of the latest Javanese Grammar as the basis for making the omission rule of Java

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Ngoko language. In addition to Mardiwarsito and Harimurti K in his book entitled "Struktur Bahasa Jawa" (Structure of Java Language) 2006.

REFERENCES :

[1] Hassan, A.K.A. and Hadi, M.J., "Proposed MABC-SDAIR Algorithm for Sense-Based Distributed Arabic Information Retrieval", *Journal of Theoretical and Applied Information Technology (JATIT)*, Vol. 95, No. 3, pp. 543-353, 2017

[2] Moral, C. Antonio, A. Imbert, R. And Ramirez, J. "A Survey of Stemming Algorithma in Information Retrieval", *Information Research*, Vol. 19, No. 1, 2014.

[3] Gupta, V. Joshi, N. Mathur, I. "Design & Development of Rule Based Inflectional and Derivational Urdu Stemmer 'Usal'", *1st International Conference on Futuristic Trend in Computational Analysis & Knowledge Management*, IEEE, 2015.

[4] Meitei, S.P. Purkayastha, B.S. Devi, H.M. "Development of a Manupuri Stemmer : A Hybrid Approach", *International Symposium on Advanced Computing and Communication (ISACC)*, 2015.

[5] Abuata, A. and Al-Omari, A. "A Rule-based Stemmer for Arabc Gulf Dialect", *Journal of King Saud University-Computer and Information Sciences*, Vol. 27, pp. 104-112, DOI.0.1016/j.jksuci.2014.04.003, 2015

[6] Mezher, K. and Omar, N. "A Backpropagation Neural Network to Improve Arabic Stemming", *Journal of Theoretical and Applied Information Technology (JATIT)*, Vol. 82, No. 2, pp. 385-394, 2015

[7] Ismailov, A. Jalil, M.M.A. Abdullah, Z. Rahim, N.H.A "A Comparative Study of Stemming Algorithms for Use With the Uzbek Language", *3rd International Conference on Computer and Information Sciences (ICCOINS)*, IEEE, 2016.

[8] Suryadi, M. Subroto, H.D.E. Marmanto, S. "The Use of Krama Inggil (Javanese Language) in Family Domain at Semarang and Pekalongan Cities", *International Journal of Linguistics*, Vol. 6, No. 3, pp. 243-256, 2014

[9] Asian, J., Williams, H. E. & Tahaghoghi, S. M. "Stemming Indonesian". *Proceedings of the 28th Australasian conference on Computer Science - ACSC '05*, **38**, pp.307-314, Australian Computer Society, Inc, 2005.

[10] Korenius, T., Laurikkala, J., Järvelin, K. & Juhola, M. "Stemming and lemmatization in the clustering of Finnish text documents". In *Proceedings of the thirteenth ACM International Conference on Information and Knowledge Management*, pp. 625-633, 2004.

[11] Majumder, P., Mitra, M. and Pal, D. "Bulgarian, Hungarian and Czech stemming using YASS". In Carol Peters, Valentin Jijkoun, Thomas Mandl, Henning Müller, Douglas W. Oard, Anselmo Peñas, Vivien Petras & Diana Santos (Eds.), *Advances in multilingual and multimodal information retrieval: 8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budpest, Hungary, September 2007. Revised selected papers*, pp. 49-56, 2008.

[12] Orengo, V. and Huyck, C. "A stemming algorithm for the Portuguese language". In *Proceedings of the Eighth International Symposium on String Processing and Information Retrieval 2001 (SPIRE 2001)*, pp. 186-193, IEEE Computer Society Press, 2001.

[13] Dinçer, B.T.,"Stemming in agglutinative languages: a probabilistic stemmer for Turkish". In A. Yazıcı and C. Sener, (Eds.), *Computer and Information Sciences - ISCIS 2003*, pp. 244-251, 2003.

[14] Eryiğit, G. C. and Adalı, E." An affix stripping morphological analyzer for Turkish". In M. H. Hamza, (Ed.), *Proceedings of the International Conference on Artificial Intelligence and Applications - AIA'04*, pp. 299-304, 2004.

[15] Braschler, M. and Ripplinger, B. "How effective is stemming and decompounding for German text retrieval?", *Information Retrieval, 7*(3-4), pp.291-316, 2004.

[16] Moulinier, I., McCulloh, J. A. and Lund, E. West Group at CLEF 2000 : "Non-English monolingual retrieval". In *Revised Papers from the Workshop of Cross-Language Evaluation Forum on Cross-Language Information Retrieval and Evaluation*, pp. 253-260, 2001.

[17] Adam, G., Asimakis, K., Bouras, C. and Poulopoulos, V. "An efficient mechanism for stemming and tagging: the case of Greek language". In Rossitza Setchi, Ivan Jordanov, Robert J. Howlett and Lakhmi C. Jain , (Eds.), *Proceedings of the 14th*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

International Conference on Knowledge-based and Intelligent Information and Engineering Systems, Part III, 389-397, 2010.

[18] Larkey, L.S., Ballesteros, L. and Connell, M. E. "Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis". In *Proceedings of the 25th annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 275-282, 2002.

[19] Taghva, K., Elkhoury, R. & Coombs, J. "Arabic stemming without a root dictionary". In *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05), Volume I*, pp. 152-157, IEEE Computer Society, 2005.

[20] Al-Shalabi, R. Kanaan, G. Ghwanmeh, S. Nour, F.M. "Stemmer Algorithm for Arabic Words Based on Excessive Letter Locations", IEEE, 2008.

[21] Manning, C.D. Raghavan, P. Schultze, H. "An Introduction to Information Retrieval, Cambridge (UK), 2008.

[22] Ali, M., Khalid, S., Saleemi, M.H., Iqbal, W., Ali, A. and Naqvi, G., "A Rule based Stemming Method for Multilingual Urdu", *International Journal of Computer Application,* Vol.134, No.8, pp 10-18, 2016.

[23] Pandey, P., Amin, D. and Govilkar, S., "Rule based Stemmer Using Marathi WordNet for Marathi Language", *International Journal of Advanced Research in Computer and Communication Engineering*, Vol.5, Issue 10, pp. 278-282, 2016.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

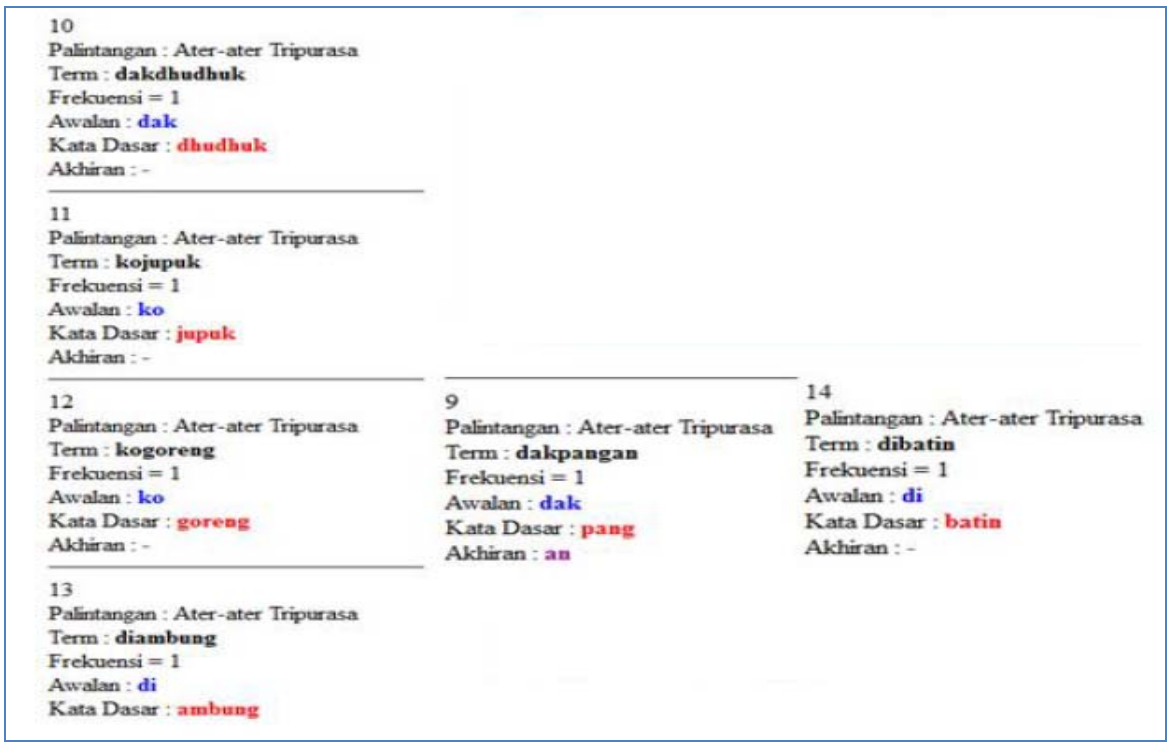

Figure 3. Sample Output of the System

Tabel 5:Calculating of Stemming Results

	Doc 1	Doc 2	Doc 3	Doc 4	Doc 5	Doc 6	Doc 7	Doc 8	Doc 9	Doc 10
Recall	0.333	0.331	0.300	0.344	0.333	0.400	0.253	0.333	0.421	0.391
Precision	0.811	0.877	0.851	0.801	0.811	0.862	0.651	0.811	0.862	0.822
F-Measure	0.473	0.480	0.443	0.481	0.472	0.581	0.445	0.472	0.581	0.498
Accuracy	0.654	0.671	0.648	0.683	0.654	0.701	0.640	0.654	0.701	0.683

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

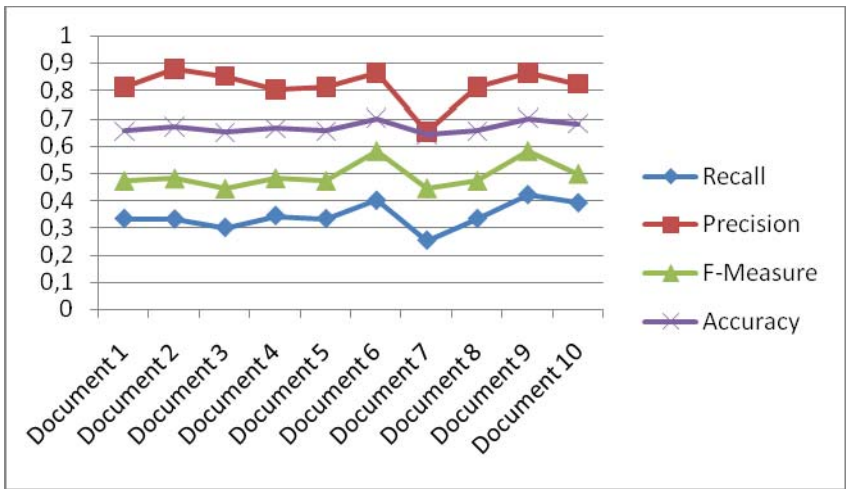

Figure 4. Testing of Stemming Results

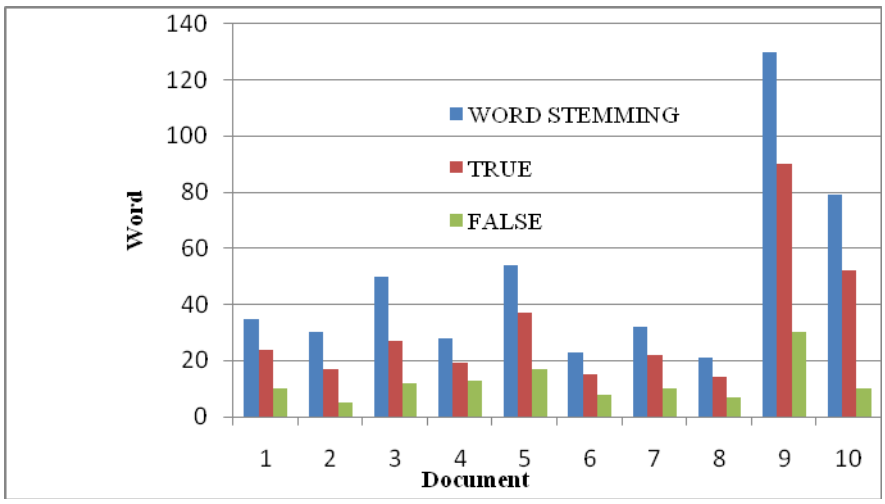

Figure 5. Distribution of Stemming Results