International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-4, February 2020

Development and Integration of an Odia Stemmer in Dspace for Odia Search Engine

Gouranga Charan Jena, Siddharth Swarup Rautaray

Abstract: ***Stemmer is used for reducing inflectional or derived word to its stem. This technique involves removing the suffix or prefix affixed in a word. It can be used for information retrieval system to refine the overall execution of the retrieval process. This process is not equivalent to morphological analysis. This process only finds the stem of a word. This technique decreases the number of terms in information retrieval system. There are various techniques exists for stemming. Here a new hybrid stemmer has developed named as "Mula" for Odia Language. It is a combination of brute force and enhanced suffix strippingapproach for Odia language. The new born stemmer is both computationally inexpensive and domain independent. We have integrated this stemmer in existing Dspace for Odia text retrieval System. The results are commendable and suggest that the new stemmer can be used effectively in Odia Search Engine. The proposed stemmer also handles over-stemming and under-stemming effectively.***

Keywords:Derivational Suffixes, inflectional words,Odia Stemmer, Information Retrieval, Brute force, DSpace etc.

I. INTRODUCTION

Stemming is atechnique for removing inflectional orderived wordto its stem.This technique is used to remove the suffix or prefix affixed in a word.This process finds stem of a word. Stemmer is essential for information retrieval system to refine the performance of the system. It technique is not equivalent to morphological analysis. Its primary objective is to decreases the number of terms in an information retrieval system. Stemming techniquecan be used in information retrieval to decrease as many related words to a common form that is not in base form. For example, the English word "Computation" has different inflections such as 'Comput', 'Compute, 'Computing', 'Computes' etc. In this case stemmer can be used to reduce derived words into its root or stem word. Many stemmers had been developed for different languages, which reduce a word to its root/stem form.It ultimately reduces the indexfile size in an information retrieval system. In this way we can improve recall (i.e. the number of documents retrieved in response to a query.) of an IR system by effectively using stemmer in the background. Since many derivational words are mapping into one word i.e. root or stem. It ultimatelyreduces the volume of the index files in the IR system. There are several types of stemming algorithms exists and it differs in respect to their performance and accuracy.

Revised Manuscript Received on February 28, 2020.
***** Correspondence Author
Gouranga Charan Jena*, Anand Pani School of Computer Engineering, KIIT DU, Odisha, India. Email-jenagouranga2000@gmail.com
Siddharth Swarup Rautaray, School of Computer Engineering,KIIT DU, Odisha, India. Email-siddharthfcs@kiit.ac.in

There are various algorithms used to find a stem of a word i.e. (a) Brute-force algorithms: It uses a lookup table that contains derived words with their corresponding roots.

To find the root/stem of a word, the table is queried to find a matching inflection word. If a matching inflection word is found, thencorresponding root returned. Otherwise, it fails. (b) Suffix-stripping Approach: It does not depend on any lookup table that consists of derived or inflected words and their root word relations. It simply uses a set "rules" which drives the algorithm. It finds the root/stem of the given input word based on that rules. (c) Lemmatization algorithms: This technique also called as text normalization. In Lemmatization root word is called Lemma. The POS is first identified of that language and then an attempt will be made to find the stem. The stemming rules will change based on a word's POS of that language. Lemmatization process ensures that the root or stem of the inflected words belongs to that language (d) Stochastic algorithms: It is based on probability method to detect the root form of a word. This trained on a table of root words to inflected words relations to develop a probabilistic model. (d) Affix Removal Approach: The name clearly suggests this approach is related to removing the suffixes or prefixesof a word. An Affix may be a prefix or a suffix. It comes under truncating method of the stemming algorithm. We found affixes are connected with nouns in Odia language. One can opt any of the above technique while designing stemmer. (e) Hybrid approach: This technique combines more than two methods as discussed above. It may merge the rule-based technique along with the probabilitymethod. (f) N-Gram Modeling: Many stemming methods used in the n-gram technique of a word to select the correct stem for a word. Stemming plays a vital role to handle the vocabulary mismatch problem of anIR system. In this said problem, the query words mismatchwith the document words. For example, when a user input a query word and the word does not existin the vocabulary of the document then it may cause unreliable result. To avoid this problem we have developed a new web-based hybrid stemmer using brute force with enhanced suffix stripping algorithm union that can be adopt in the Odia information retrieval system. The new stemmer is both computationally fasteras well as domain-independent.

II. RELATED WORKS

In the study of information retrieval, researchers find stemming plays an important role. Stemming is not a new concept. Stemming techniques had invented since 1968. The first stemming algorithm was designed by Julie Beth Lovins[1]. After that many researchers continued investigating various approaches to this area of study and proposed several algorithms to improve its performance.

Another stemmer in English was written by Martin Porter [2] in the year 1980. As compared to European languages as well as English,

a fewresearcheshave been discovered in Indian Language.

A Hindistemmer [3] was proposed by Rao, Durgeshet al. based on suffix striping approach.

A Bengali Morphological analyzer [4] was developedby Dasguptaet al. based on suffix striping approach. Stemming is the process by which the user inputs an inflected word to the trained model and the model produces the root/stem word according to its rule set. In this Paper we havedeveloped a Stemmer based on Hybrid Approach.

III. LITERATURE SURVEY ON ODIA STEMMER

Reference	Key Findings
Sampa et al. [5]	Publisheda paper on Stemmer for Odia language. They used the suffix stripping approach to remove the inflectional suffixes. The limitation of this algorithm was it only predicts 88% accuracy.
Balbantray, R.C. et al. [6]	Published a paper FIRE 2012 Submission: MET Track Odia. They had used the affix removal algorithm. The system reads input text files from the folder. Firstly, it removes stop words from the input files against the stop word dictionary then matched the token with the root word dictionary. After that the input matched the suffix dictionary then removes the suffix and match with the root word. If the root word found then there is no further processing required.
Balabantaray, R.C. et al. [7]	Presented a paper on Odia Text Summarization.
Sethi, Dhabal Prasad [8]	Published a paper onLightweight Stemmer for Odia Derivational Suffix. He used suffixstripping method to find the stems.

IV. ODIA DERIVATIONAL MORPHOLOGY

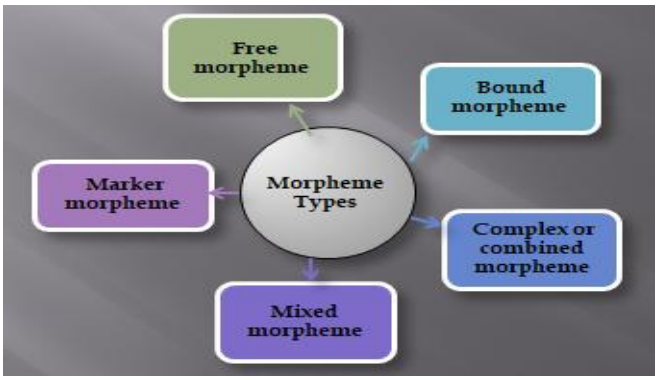
a) Odia Morphology:

The formal variants of a morpheme are called allomorphs of that morpheme. The variant may be phonologically or morphologically conditioned. A morpheme may be a free or a bound form. Alternatively we can say that a word consist of one or more than one morpheme. From the point of view of its internal structure, a word may consist of (i) a root morpheme only (ii) a root and one or more non root morpheme or (iii) more than one root morpheme. The non-root morphemes are bound forms and are generally referred to as affixes. Roots enter into further morphological constructions and form a base while non-roots do not. [9]

Odia morphology deals with the analysis, identification and description of structure of morpheme. Morphology deals with the structure of words. The basic unit is the focus of study in morphology is morpheme.

Example: The word ବାଳକମାନେ the morphemes are ବାଳକ, ମାନେ. Morpheme is not always conveying a meaningful word in Odia. Any morpheme in Odia should be a root word, prefix or suffix. Morphemes are divided into five categories shown in fig 1.

Fig. 1.Types of Morpheme



The morpheme which are independent called free morpheme. Those morphemes are standalone in nature. It does not need to add with other to create a word.Examples offree and bound morpheme discussed below.

ରାମ ଭାତ ଖାଉଛି । ରାମ ଭାତ(କୁ) ଖାଉଛି|

The morpheme ଭାତis a stand-alone morpheme and morpheme (କୁ) is a suffix. Most of the morphemes are bound type in Odia language.

An affix is a morpheme that may come at the beginning (Termed as Prefix) or the end (Termed as Suffix) of a base morpheme.In Odia, prefixes are bound morphemes are affixes that come before a base morpheme. For example:

/ଉପକୂଳ/ = /ଉପ/ + /କୂଳ/

b) Odia Derivational Morphology:

Odia morphemes of different types such as nouns, verbs, affixes, etc. combine to form new words. The suffixes can added with root word to form POS shownin the below Table- I.

Categories	Examples
Noun to Adjective	Noun word + Derivational suffix =Adjective category. ରୂପ+ଏଲି = ରୂପଲେି
Adjective to Noun	Adjective word + derivational suffix = noun words ଆଧୁନିକ + ତା=ଆଧୁନିକତା
Adjective to Adjective	Adjectiveword + Suffix= Adjective word ସରୁ+ଆ=ସରୁଆ
Verb to Adjective	Verbal word + Derivational suffix =Adjective word ପାନ+ଇଆ=ପାନିଆ
Verb to Noun	Verbal word+Derivational suffix= Noun word. ବିଚ୍+ଅଣା=ବିଚ୍ଚଣା ଖଳେ+ଆଳି=ଖଳୋଳି

In the suffix-stiffing approach of stemming, we remove suffixes or affixes attached to that stem.As Odia is being influenced by Sanskrit. So Odia language has strong inflectional system.

According to the Panini grammar rules, we are representing some rule in below example.

For example 'କଲମଗୁଡ଼ିକ' here stem କଲମ and suffix is ଗୁଡ଼ିକ .

The nominal suffix and verbal suffix of Odia language are described below. (Table – 1&2) [10].

Table- II: List of Nominal Suffixes in Odia

ବିଭକ୍ତି Inflection	ଏକ ବଚନ (Singular)	ବହୁ ବଚନ (Plural)	କାରକ (Case-Relationship)	ଉପପଦ (Noncase-Relationship)
ପ୍ରଥମା (1st Inflection)	ଏ,ଟ,ଟି,ଟା,ଟିଏ,ଟାଏ,ଃ	ଏଗୁଡ଼ିଏ,ଗୁଡ଼ାକ,ଗୁଡ଼ିକ,ମାନ ,ମାନ ,	କର୍ତ୍ତା	
ଦ୍ୱିତୀୟା (2nd Inflection)	କୁ,ତ,ଟିକୁ,ଟାକୁ,ଙ୍କୁ,କି,ଙ୍କୁ, ଠାକୁଠିକି,ଠିକୁ,	ଙ୍କୁଗୁଡ଼ାକୁ,ଗୁଡ଼ିକୁ ,ମାନଙ୍କୁ ,	କର୍ମ	
ତୃତୀୟା (3rd Inflection)	ରଙ୍କେଦ୍ୱାରା,ଦ୍ୱେ,ଦ୍ୱାରା,	ଙ୍କଦ୍ୱାରା ,ଦ୍ୱେ ମାନଙ୍କ ,ଦ୍ୱାରା ମାନଙ୍କ , ମାନଙ୍କରେ	କରଣ	
ଚତୁର୍ଥୀ (4th Inflection)	କୁ,ଙ୍କ,ଲାଗି,ପାଇଁ,ଙ୍କି,କି,ଙ୍କୁ, ନିମନ୍ତେ ଙ୍କ ,ନିମନ୍ତେ ଙ୍କ ,ଲାଗି ଙ୍କ, ସକାଶେ ଙ୍କ ,ସକାଶେ ,ଯେ·ାଗୁଁ	ଙ୍କୁ ,ଲାଗି ମାନଙ୍କ ,ଗୁଡ଼ାକୁ,ଗୁଡ଼ିକୁ ,ମାନଙ୍କୁ , ନିମନ୍ ,ମାନଙ୍କନିମନ୍ତତେଯେ·ାଗୁଁ ମାନଙ୍କ, ସକାଶେ	ସମ୍ପ୍ରଦାନ	
ପଞ୍ଚମୀ (5th Inflection)	ରୁଠଉଁ ,ଠୁଁ ,ଠାରୁ ,	ମାନଙ୍କଠାରୁ,ମାନଙ୍କଠୁଁ , ଙ୍କଠାରୁଠୁ ମାନଙ୍କ ,ଠଉଁ ମାନଙ୍କ ,	ଅପାଦାନ	
ଷଷ୍ଠୀ (6th Inflection)	ର,ଠି,ଠାଇଁ ,ଏ ,ଙ୍କର ,ଙ୍କ , ଠଙ୍କେଁଠି ,ତହିଁ,	ମାନଙ୍କଙ୍କରି ,ମାନଙ୍କରି ,ମାନଙ୍କର ,		ସମ୍ବନ୍ଧ
ସପ୍ତମୀ (7th Inflection)	ରେ,ଠେଇଁ,ଠି,ଠାଇଁ,ଏ,ଠାରେ, ତହିଁଠି ,	ମାନଙ୍କରେ ,ଠାଇଁ ମାନଙ୍କ ,ମାନଙ୍କଠାରେ, ଠଇଁ ମାନଙ୍କ ,ଠି ମାନଙ୍କ	ଅଧିକରଣ	

Table- III: Odia Verbal Suffix

କାଳ (Tense)	ପୁରୁଷ (Person)	ଏକ ବଚନ (Singular Suffix)	ବହୁ ବଚନ (Plural Suffix)
ବର୍ତ୍ତମାନ କାଳ (Present Tense)	ପ୍ରଥମ ପୁରୁଷ	ଉଅଛି,ଉଛି,ଉଥାନ୍ତି,ଏ	ଉଛୁ, ଇଅଛୁ, ଉଥାନ୍ତୁ, ଉ
	ଦ୍ୱିତୀୟ ପୁରୁଷ	ଉଅଛୁ, ଉଛୁ, ଉଥାନ୍ତୁ, ର	ଉଅଛ, ଉଛ, ଉଥାନ୍ତ
	ତୃତୀୟ ପୁରୁଷ	ଉଅଛି, ଉଛି, ଉଅଛନ୍ତି, ଉଥାନ୍ତା, ଉଥାନ୍ତେ, ଉଥାଆନ୍ତେ, ନ୍ତି, ଉଆନ୍ତା	ଉଛନ୍ତି, ଉଅଛନ୍ତି, ଉଥାଆନ୍ତି, ନ୍ତି
ଅତୀତ କାଳ (Past Tense)	ପ୍ରଥମ ପୁରୁଷ	ଇଲି,ଇଛି, ଇଥିଲି, ଥିଲି,ଇଥାନ୍ତି,ଇଅଛି	ଗଲୁ, ଇଥିଲୁ,ଉଥିଲୁ, ଇଥାନ୍ତୁ
	ଦ୍ୱିତୀୟ ପୁରୁଷ	ଇଲୁ,ଇଛୁ, ଇଥିଲୁ, ଉଥିଲୁ, ଇଥାନ୍ତୁ	ଇଲ, ଇଛ, ଇଅଛ, ଇଥିଲ, ଉଥିଲ, ଇଥାନ୍ତ
	ତୃତୀୟ ପୁରୁଷ	ଇଲା, ଇଲେ, ଇଛି, ଇଛନ୍ତି, ଇଥିଲେ, ଇଥିଲଲ, ଉଥିଲେ, ଇଥାଆନ୍ତା, ଇଥାଆନ୍ତେ	ଇଲେ, ଇଛନ୍ତି, ଇଥିଲେ, ଉଥିଲେ, ଇଥାଆନ୍ତେ
ଭବିଷ୍ୟତ କାଳ (Future Tense)	ପ୍ରଥମ ପୁରୁଷ	ଇବି, ଇଥିବି, ଥାନ୍ତି, ଉଥିବି	ଇବୁ, ଇଥିବୁ, ଥାଆନ୍ତୁ, ଉଥିବୁ, ଇବା
	ଦ୍ୱିତୀୟ ପୁରୁଷ	ଇବୁ, ଇଥିବୁ, ଥାଆନ୍ତୁ, ଉଥିବୁ	ଇବ, ଇଥିବ, ଥାଆନ୍ତୁ, ଉଥିବ
	ତୃତୀୟ ପୁରୁଷ	ଇବ, ଇବେ, ଇଥିବେ, ଇଥିବ, ଥାଆନ୍ତ, ଥାଆନ୍ତେ, ଉଥିବ, ଉଥିବେ	ଇବେ, ଇଥିବେ, ଥାଆନ୍ତେ, ଇଥିବେ

There are some prefixes which are attached to noun only in Odia language. There is20 such prefixes exist in Odia. These are inherited from Sanskrit. These are as follows (Table -3):

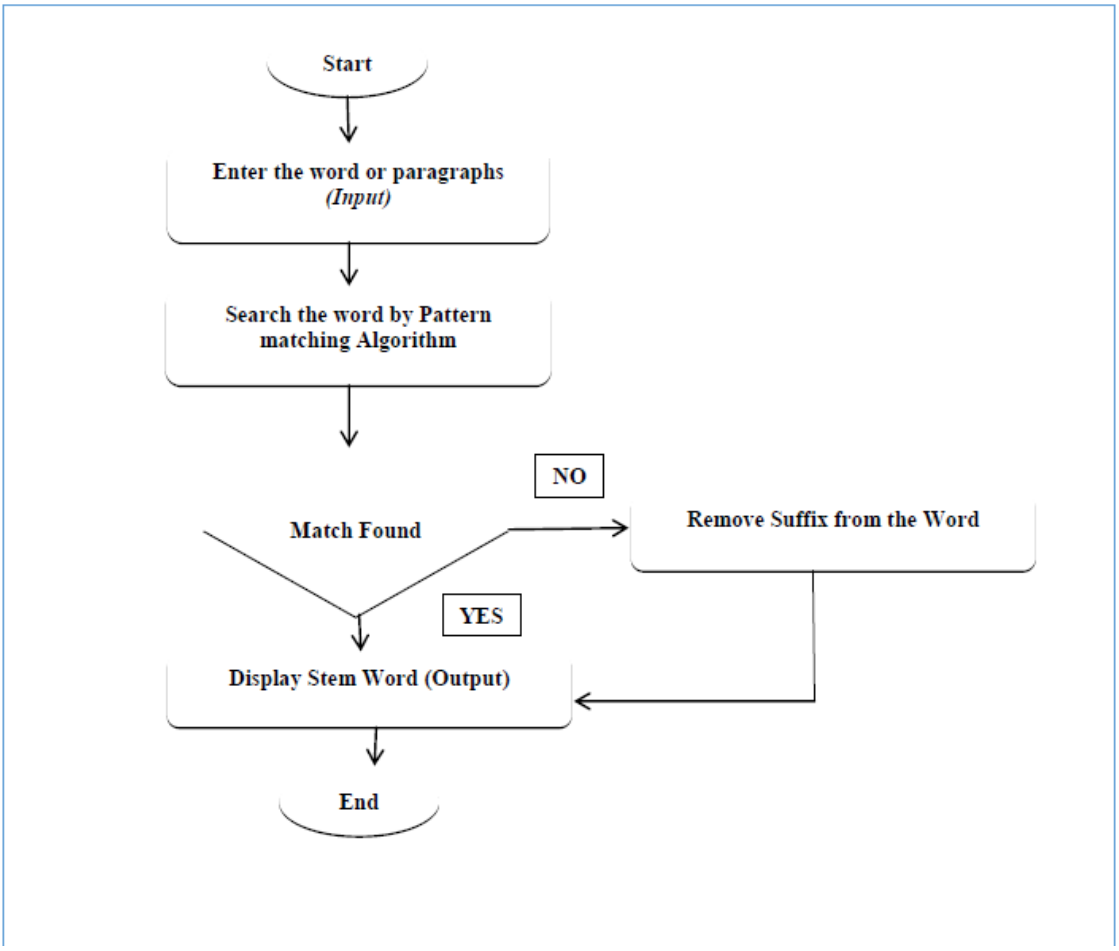
Table- IV: List of Odia Prefixes

ପ୍ର	ପରା	ଅପ	ସମ୍	ନ
ଅଧି	ସୁ	ନିର୍	ଉତ୍	ପରି
ପ୍ରତି	ଅବ	ଅନୁ	ଦୁର୍	ବି
ଅଭି	ଅତି	ଅପି	ଉପ	ଆ

V. PROPOSED METHODOLOGY FOR ODIA STEMMER

We have proposed a new web-based stemmer based on hybrid approach. [11][12] for Odia language. The new born stemmer is both computationally faster and domain independent. The algorithm of the proposed stemmer is described in the below flowchart. We developed a stemmer algorithm for Odia which removes derivational suffixes from derived word. The algorithm is a combination of brute force approach and enhanced suffix stripping approach.In this new enhanced model of suffix removal technique, system remove the suffixes recursively first 3 character, then 2 characters and last 1 character from the derived words.

Fig.2. Flowchart of Proposed Odia Stemmer Algorithm



Brute force search is called as exhaustive search. It searchesall the possible solution from the data. Here it searches the root words present in the database.This techniqueuses a lookup table which contains inflected words and root words mapping. In this technique [13] we create and store maximum possible inflected words along with their corresponding root word in a database table. When we givesinput to the system then brute force search is carry out and it inspects that whether the derivational wordsexist in the database. If the word is presentin thattablethen it will give its corresponding stem or root word. If the word is not present in the table then it will go for suffix removalmethod to handle those words. Suffix removal is a rule based approach inthat certain rule set is defined. By applying those rule set suffixes are removed from the inflected or derived word, to find the stem/root. [14] The new enhanced approach of suffix stripping algorithm is:

Start

Step 1: Enter derivational word that to be stemmed

Step 2: The system removes the 3 character suffixes,2 character suffixes and 1 character suffixesfrom the derivationalword if word length greater than three, and two respectivelyrecursively.

End

The inflected word is processed by the stemmer in three steps. The steps are shown below.

Input: The inflected Odia word/paragraph is entered as an input to the web-based system. Here "ଆଧୁନିକାତା" is given as an Input word.

Processing: Derivational/inflected word is searched by brute force method. It matches with the user searched word with the words exist in the database table. If the matching word is existin the database then it will provide the stem of the word as output. If mismatch found then it search for the alternate methodcalled suffix stripping method i.e. the algorithm removes the suffixes recursively first 3 characters, then 2 characters and last 1 character with a condition that the inflected word must be greater than the suffix to find the stem/root of the word.

One Character derivational Suffix :
ଅ,ଆ,ଇ,ଈ,ତ,ତି,ଯ,ତ୍ର,ତ୍ରୁ,ନ,ତା,ତ୍ୟା,ଦା,ଧା,କା,ରା

Two Character derivational Suffix:ଅନ,ମସ୍ତ,ଅକ,ତର,ତକ,ତମ,ଇନ,ଇଷ୍ଟ,ଇଷ୍ଣୁ,ଆଳୁ,ତବ୍ସ୍ତ,ଆଳ,ଇତ୍ର,ଆଳି,ତର,ଆରି,ସନ,ଇଆ,ଅଣ,ତଆ,ଅଣା,ତରା,ଅନି,ଏଲି,ଅନ୍ତା,କାର,ଅନ୍ତୀ,କୁଳା,ଆଣ,ବନ୍ତ,ଅତ,ତିଆ,ତଆ,ପଣ,ତଣି,ଖାନା,ତଣା,ଖତ·ାର,ଏଣି,ଗର,ଇକ,ଗିରି,ଇସ୍ତ,ଆମି,ଇନ,ବାଜ,ଏସ୍ତ,ବାଲା,ଇତ,ଦାର,ଇଳ,ବିନ,ଇନ,ତନ,ବତ**Three Character derivational Suffix**: ଅନୀସ୍ତ, ତଆଳ, ତଆଳି,ଗାଳିନ,ଇସ୍ତସ,ସ୍ଥାନୀସ୍ତ,ଆଳିଆ,

Output: The output after processing is "ଆଧୁନିକ."

VI. RESULT & DISCUSSION

We have evaluated the stemmer by taking different set of words i.e. 100 words, 200 words, 300 words and so on to calculate the time taken to extract the root words. We have not compared our Odia stemmer with any of the existing stemmer available for Odia language. Nowhere had we found the existing result to compare with the proposed stemmer.

Table 4: Time Taken to Extract Odia Root Words

Set No	No of Words	Time Taken (Sec.)
Set-1	100	5.07
Set-2	200	8.92
Set-3	300	13.27
Set-4	400	17.18
Set-5	500	20.29
Set-6	600	24.84
Set-7	700	29.11
Set-8	800	32.85
Set-9	900	37.66
Set-10	1000	40.48

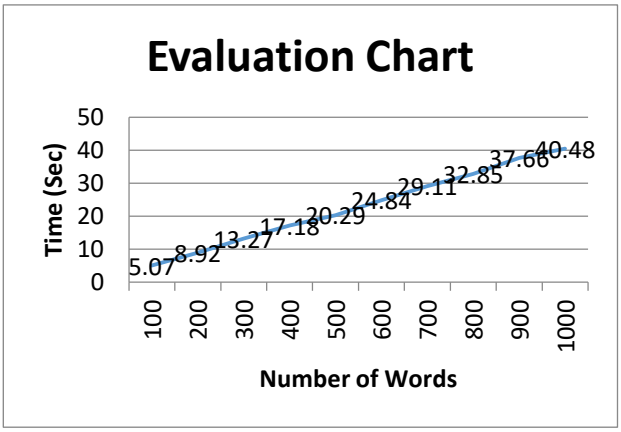


Fig. 3. Evaluation graph

VII. CONCLUSION AND FUTURE WORK

This stemmer can be used effectively to improve the query translation performance for information retrievalsystem [15]. Finally, we have integrated the new stemmer in the existing Dspace for Odia text retrieval system. The results are commendable and implicates that this can be used effectively in Odia Search Engine. It also handles the problem of under stemming and over stemming in some extent. In future we can integrate this stemmer for Odia text summarization purpose. As a future scope this system can be enhanced by including some more inflected word and corresponding to their stem mapping in to the database.

ACKNOWLEDGMENT

We are thankful to all the Language Researcher who directly and indirectly help to design and develop this web based Odia Stemmer.

REFERENCES

1. Lovins, J.B.,"Development of a stemming algorithm", Mechanical Translation and Computational Linguistics,1968, Vol. 11 Nos 1 and 2, pp. 22-31.
2. Porter, M.F. (1980), "An algorithm for suffix stripping",1980, Program, Vol. 14 No.3, pp.130-137.
3. Ramanathan A, Durgesh D Rao, "A lightweight stemmer for Hindi", National Center for Software technology.
4. Dasgupta,S., V. Ng.,"Unsupervised morphological parsing of Bengali",Lang Resources &Evaluation,Human Language Technology Research Institute, University of Texas at Dallas,Richardson, TX 75083, USA.
5. Chaupattnaik,S.,Nanda,S.S.,Mohanty,S.,"A suffix stripping Algorithm for Odia Stemmer", Int. Journal of Computational Linguistics and Natural Language Processing,2012.
6. Balabantaray,R.C.,Sahoo,B.,Swain,M.,Sahoo,D.K.,"IIIT-BH FIRE 2012 Submission: MET Track Odia".
7. Balabantaray,R.C.,Sahoo,B.,Swain,M.,Sahoo,D.K.,"Odia Text Summarization using Stemmer",Int. Journal of Applied Information Systems (IJAIS),Foundation of Computer Science FCS, New York,USA,Volume 1– No.3, February 2012
8. Sethi,D.P.,"Design of Lightweight Stemmer for Odia Derivational Suffixes",International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 12, December 2013.
9. Odia grammar book of class 9th in BSE, ODISHA.
10. Chaupattnaik,S.,Nanda,S.S.,Mohanty,S.,"A suffix stripping Algorithm for Odia Stemmer", Int. Journal of Computational Linguistics and Natural Language Processing,2012.
11. Mohammad,A., Oqeili,S., and Rawan A. A.,"Occurrences Algorithm for String Searching Based on Brute-force Algorithm", 2006 Jordan Journal of Computer Science Vol No.2,Issue No.1,pp 82-85.
12. Mishra U. et al., "MAULIK: An Effective Stemmer for Hindi Language", International Journal on Computer Science and Engineering (IJCSE).
13. Erritali,M., "Information Retrieval: Textual Indexing Using an Oriented Object Database",Indonesian Journal of Electrical Engineering and Computer Science, Vol. 2, No. 1, April 2016, pp.205~214.
14. Sethi,D.P.,"Morphological Analyzer for Sambalpuri Odia Dialect Inflected Verbal Forms", International Journal of Advanced Reseach in Computer Science and Sofiware Enginnering,October,2013.
15. Bajpai1, P., Verma,P., Abbas,SyedQ.,"Two Level Disambiguation Model for Query Translation",International Journal of Electrical and Computer Engineering (IJECE), Vol. 8, No. 5, October 2018, pp. 3923-3932.

AUTHORS PROFILE

Gouranga Charan Jena, is continuing his Ph.D. in CSE at School of Computer Engineering, KIIT DU, Bhubaneswar, Odisha, India. He has completed his ME(CSE) from Utkal University, Bhubaneswar,Odisha,India.
Area of Research: NLP, image processing, information retrieval, data mining, ML and Big data technologies.

Siddharth Swarup Rautaray, is working as an Assistant Professor in School of Computer Engineering, KIIT DU, Bhubaneswar, and Odisha, India. He has completed his Ph.D. in CSE from IIIT, Allahabad.
Research Interest: HCI, NLP, image processing and Big data technologies. He has published many international journals and conference papers on above research areas.