International Journal of Engineering & Technology, 7 (2) (2018) 551-554

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i2.9504
Research paper

VBS Stemmer: A vocabulary-based stemmer

Hamed Zakeri Rad[1]*, Sabrina Tiun[1], Saidah Saad[1]

[1] Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia
**Corresponding author E-mail: jerald0000030@yahoo.com*

Abstract

Stemming is referred to a procedure of reducing all words appearing in different morphological variants to a common form. As a matter of fact, it is considered as a functional way in various areas of information-retrieval work and computational linguistics. In this paper, we introduced the Vocabulary Based Stemmer (VBS) as the alternative solution to the stemming problem for the applications which are based on the semantic relation between words or dictionary based and need valid words. The Vocabulary part of VBS stemmer is generated based on WordNet. To validate the VBS Stemmer, part of "Cranfield 1400" test collection being used, and the result shows significant improvements over the previous stemmers.

Keywords: *English Suffix Removal; Information Retrieval; Stemming Algorithm; Suffix Removal; Vocabulary Based Stemmer;*

1. Introduction

In an Information Retrieval (IR) system, queries and documents are usually written in natural language. This indicates that, even if referred to a common concept, the words may probably occur with many morphological variants. There is a basic idea in stemming saying that words that are similar in morphology are seeming to be similar in meaning. Thus, they can be addressed as equivalent [1]. A stemming algorithm as a computational procedure decreases all words which have the same root or base to a common form, usually by means of stripping each word of its derivational and inflectional suffixes [2].The base of a word in the automated morphological analysis, may not be as interesting as its suffixes which can be used as clues to grammatical structure. In order to reduce the overall number of terms in the IR system suffix stripping process is used which reduces the data size and complexity in the system, which is absolutely beneficial [3].

A stemming algorithm is used for maximizing the usefulness of the term. The information significant to the user semantically is included in the stems of the lexical words and suffixes help this information to be uttered in a grammatical form. One fact which is notably common in any retrieval systems is that the form of the words which is called as an input of a query does not correspond to the original words given in the corpus. In order to let the words in both corpus and query matched, they can be stripped of the suffixes that prevent them from matching. Take "walk" as the example: "walks", "walking", "walked" and "walker", can be stemmed to "walk". This has been labeled as desirable by many researchers who are working in many areas of computational linguistics and information retrieval.

It was Julie Beth Lovins (1968) [2], who developed the first stemming algorithm for word matching. Her approach includes a two phase stemming system. The first phase which is the stemming algorithm properly retrieves the stem of a word by means of removing its longest possible ending. The second phase handles spelling parts which are exceptions, mostly instances in which the same stem varies slightly in spelling according to the original suffixes. Her proposed stemming algorithm relies on two fundamental principles which are namely longest-match and iteration. The latter removes the whole possible suffix in the word. For instance, a word, like "relationship" consists of two suffixes "ship" and "ion" which will be removed by iteration. The former searches for the longest possible suffix match and remove it rather than removing the short suffix match. Take the word "cancellation" as an example. The longest suffix "ation" is removed not "ion".

Primarily, the stemmer of Lovins was note worthy for what it could accomplish in addition to its seminal influence on later work in this area. The algorithm design was mainly affected by the technical vocabulary with which Lovins was working. Other attempts were made to develop a stemming algorithm [4], [5], but the most well-known stemming algorithm is Porter Stemmer which was developed by F. Porter [3], [6], this algorithm was different from Lovins's in two main ways. Firstly, it is a significant reduction in the complexity of the rules associated with suffix removal and secondly, it is the use of a single, unified approach for the handling of context. There were a lot of contexts sensitive rules which were related to the length of the stem which remains after the suffix is removed [7]. The stemming algorithm paper of Porter, as a part of a larger project of IR, was originally written in 1979 in Cambridge (England). It appeared as Chapter 6 of the final project report entitled "New models in probabilistic information retrieval" [8], and it was also published by his stimulation in "An algorithm for suffix stripping" [3].

Regarding stemmers, the Paice/Husk stemmer can be named as well-known too [9]. It is an iterative algorithm which uses the same rules and suffixes in every loop. Bacchin [1] examined a stemming algorithm based on link analysis procedures. It said that prefixes and suffixes, that are stems and derivations, form communities once extracted from words. The stemmers are not only limited to English, there are stemming algorithm developed for other languages like Dutch [10], Malay [11], Portuguese [12], Hindi [13], Farsi [14], Bulgarian [15], Hungarian [16], Arabic [17] and etc.

The challenging part concerned with the conventional stemmers is the output of a stemming routine which might be ambiguous or not English. This issue can occur when a suffix is identical to the end of some root. For example, "er" is a suffix for nouns in the

Copyright © 2018 Hamed Zakeri Rad et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

"cheaper", but, in fact, it is a part of a verbal root in the "anger". Such situations bring about restrictions on the use of suffixes which play a role in determining the parts of speech. However, using the conventional stemmers to stem query and the corpus in the application in which keyword matching is used is not very complicated due to the fact that both of them can go through the stemming process and the keyword matching will be successful regardless of presentation of the words. For instance, "abut" is a stem to the word "abutter", and because in the both side (the query and the corpus), stemmed words are same so the matching process will succeed. However, these types of stemmers are not useful when the application is based on the semantic relation which exists between words in addition to using the dictionary to observe the words to process.

The most common method is called the affix removal that stemmers used for conflation. By means of this method suffix or prefix is removed from the words in order to get a common stem word. Stemming algorithms can be extensively classified into two categories [18]:

- Rule Based: A rule based approach language contains specific rules encoded. And stemming is performed based on these rules. There are different conditions specified in this approach for converting a word to its derivational stem. It must be mentioned that a list of all valid stems are proposed and there are also some rules which are exceptional are used to handle those exceptional cases.
- Statistical: Statistical stemming is referred to an effective and common approach used in information retrieval. [19], [22]. Statistical stemmers use the statistical information from a large corpus of a proposed language in order to learn morphology.

The approach of stemming taken here is based on the method of removing the affix. VBS stemmer involves two separate stemming processes. The first process of the stemming algorithm is irregular verbs, exceptions and special words, and the second process is based on the proper stemming rule implemented in the algorithm, the implementation of the vocabulary based stemmer will be explained in the next section.

2. Vocabulary based stemmer (VBS)

The method of Affix removal is based on two principles, one is iterations and the other is longest suffix match. An iterative stemming algorithm, as its name suggests, is simply a recursive procedure. It removes suffix of the word in each loop step by step. No more than one match is allowed within a single pass. Iteration is about removing all possible suffixes if a word has got more than one suffix. The longest suffix match principle says that within any given class of endings, the one which has got the longest ending should be removed first if more than one end provides a match.

As we mentioned in the introduction, the stemming algorithm built from two separate stemming processes. The first process is based on the irregular verbs, exceptions and special words. List of suffix which are implemented in the algorithm are shown in table 1.

We developed a vocabulary, based on the WordNet [23] as part of the stemmer to double check the stemmed word to ensure the successful stem. The candidate words are always the words that exist in the vocabulary, words which do not exist in the vocabulary, by default consider as special words like names.

Table 1: List of Available Suffix in the VBS Stemmer

List of available suffix in the stemmer			
'isation' and 'ization'	'ally'	'i'	'ly'
'woman' and 'man'	'ar'	'ia'	'ment'
'hood' and 'like'	'ate'	'ian'	'ness'
'ship' and 'wise'	'ative'	'ic'	'or'
'ise' and 'ize'	'cy'	'ie'	'ous'
'ant' and 'ent'	'ed'	'ing'	'ress'
'ance' and 'ence'	'ee'	'ion'	'ry'
'ies','es' and 's'	'er'	'ish'	'ty'
'able'	'est'	'ism'	'y'
'age'	'ful'	'ist'	
'al'	'fy'	'ive'	

In the candidate words, there are always words which have the same ending as the suffixes but these endings are part of the word verbal root and should not get stemmed. Using the vocabulary to check the existence of the stemmed words will help to identify some of these words, but there are always exceptional words. For example, consider the word "anger" as the candidate word, which will fall in the "er" suffix category. The word "anger" after stem will be "ang" (definition: a civilian reserve component of the united states air force that provides prompt mobilization during war and assistance during national emergencies), which exist in the vocabulary and will consider as a successful stem which is a wrong stem. To identify these kinds of words, the exceptional list is added to the stemmer which contains all the words that should not get stemmed.

There are two other lists added to the stemmer which contains irregular verbs and special words; irregular verbs list contains the list of all irregular verbs in the English language, to stem the irregular verbs all the verbs get converted to the present form of the verb. For example "sunk" and "sunken" will be converted to "sink". The special words list contains the list of words which their stem do not follow the general rule implemented in the stemmer. For example words like "absorbent" and "absorbable" will get stemmed to "absorb" but the word "absorption" or "absorptive" get stemmed to "absorp" which is not a real word and automatically consider as exceptional words and they get not stemmed. But the proper stem for "absorption" or "absorptive" is "absorb", these kinds of words are considered as special words.

There are another types of words which are in the special words list, these kinds of words get the wrong stem if they follow the implemented rules in the stemmer. For example, the rules for stemming 'er' suffix is first to remove 'er' from the end of the word, then check the vocabulary, if the word doesn't exist add 'e' at the end of the word. This is a general rule but it doesn't apply to all words. For example consider 'wager' as the candidate word, following the general rule the 'wager' get the stem to 'wag' which is in the vocabulary and consider as a successful stem, but the stem has a totally different meaning and it is not a correct stem. The correct stem for 'wager' is 'wage' which makes it a special word.

Identifying exceptional words and special words is impossible through the rules implemented in the stemmer, so as the alternative solution to the problem we ran the entire vocabulary through the stemmer and double check the words and the result stemmed words, based on the definition of them in the WordNet [23] and singled out each exceptional words and special words one by one. The special words list contains the appropriate stem for each special word. Table 2 shows the total number of words in each suffix category and the total number of exceptional words.

Table 2: Total number of words in each suffix category and the total number of exceptional words

Total Candidate Number: 78051		
Suffix Category	Total Words	Exceptional Words
'isation' and 'ization'	536	14
'ise' and 'ize'	1227	276
'ant' and 'ent'	2459	765
'ance' and 'ence'	978	207
'ies', 'es' and 's'	12651	4414
woman	71	13
ative	293	73
hood	47	2
like	259	15
man	555	160
ship	235	14
wise	31	10
ment	884	138
ally	662	11
ress	226	79
ness	2146	33
able	1002	475
ate	1786	1137
ive	589	293
ion	3709	1044
ful	253	45
ism	868	240
age	685	175
ish	632	241
ist	876	141
ous	996	644
ian	693	265
ing	3964	431
est	319	91
ly	3179	327
ty	1803	602
ic	2376	1246
ar	1163	457
or	1436	306
er	8150	1942
al	2771	1181
ia	2904	1392
cy	437	86
ee	795	172
ry	2072	727
fy	170	110
ie	340	130
ed	4857	1133
i	1807	757
y	4159	1655

The second part of the stemming process is to stem the words in each suffix category, the words which are not part of the irregular verbs, exceptional words and special words list. These words can be correctly stemmed according to the stemming rules implemented in the stemming algorithm (These are the words which got correctly stem during the definition check to find exceptional words and special words). This part of the algorithm has five iterations to remove all the possible suffixes from the word. In case the word has more than one suffix. After each iteration and suffix removal, the exceptional words list and special words list get checked to avoid over stemming. For example, consider word "archives" as the candidate word. In the first iteration "archives" get the stem to "archive", in the second iteration "archive" gets the stem to "arch" which is a wrong stem and it's an over stemming. This makes the word "archive" an exceptional word and gets identified by checking the list after iteration.

3. Result and analysis

The aim of the experiment is to compare the output result of vocabulary based stemmer which explained in the previous section, to other well known stemmer. For this experiment, we use the latest online version of Porter stemmer. For the comparison, we use the first 10 documents of "Cranfield 1400" (available online at: http://ir.dcs.gla.ac.uk/resources/test_collections) test collection against the latest online version of Porter [3]. The "Cranfield 1400" test collection is a well known test collection for Information Retrieval. It is not possible to include all the first 10 documents in this paper so we only add the first document and the result for other documents will be shown in table 3.

Original Document Text: "Experimental investigation of the aerodynamics of a wing in a slipstream. An experimental study of a wing in a propeller slipstream was made in order to determine the spanwise distribution of the lift increase due to slipstream at different angles of attack of the wing and at the different free stream to slipstream velocity ratios. The results were intended in part as an evaluation basis for different theoretical treatments of this problem. The comparative span loading curves, together with supporting evidence, showed that a substantial part of the lift increment produced by the slipstream was due to a destalling or boundary-layer-control effect. The integrated remaining lift increment, after subtracting this destalling lift, was found to agree well with a potential flow theory. An empirical evaluation of the destalling effects was made for the specific configuration of the experiment."

Porter Stemmer Result: "experiment investig of the aerodynam of a wing in a slipstream An experiment studi of a wing in a propel slipstream wa made in order to determin the spanwis distribut of the lift increas due to slipstream at differ angl of attack of the wing and at differ free stream to slipstream veloc ratio the result were intend in part as an evalu basi for differ theoret treatment of thi problem the compar span load curv togeth with support evid show that a substanti part of the lift increment produc by the slipstream wa due to a destal or boundari layer control effect The integr remain lift increment after subtract thi destal lift wa found to agre well with a potenti flow theori An empir evalu of the destal effect wa made for the specif configur of the experi."

Vocabulary Based Stemmer (VBS) Result: "Experiment investigate of the aerodynamic of a wing in a slipstream. an experiment study of a wing in a propel slipstream be make in order to determine the span distribute of the lift increase due to slipstream at differ angle of attack of the wing and at differ free stream to slipstream velocity ratios. the result be intend in part as an evaluate base for differ theoretic treat of this problem. The compare span load curves, together with support evidence, show that a substantial part of the lift increment produce by the slipstream be due to a destalling or boundary-layer-control effect. The integrate remain lift increment, aft subtract this destalling lift, be find to agree well with a potential flow theory. An empiric evaluation of the destalling effect be make for the specify configuration of the experiment."

3.1. Analysis

In the above comparison test, there are 3 unrecognized words "destalling" (based on vocabulary) in the original document text. In the porter stemmer result, the number of unrecognized words increased to 37 but in VBS Stemmer result, the number of unrecognized words remains at 3 (VBS stemmer never creates an unrecognized word, simply because of using the WordNet based vocabulary to identify the words. For VBS Stemmer, all unrecognized words mentioned in the table 3 already existed in the original document text). In the original document text, VBS stemmer recognized 28 special words which should not get stemmed, such as "aerodynamic". And porter stemmer recognized 9 special words such as "increment" and "differ". In the same manner, the comparison results for the first 10 documents of Cranfield 1400 are shown in the table 3.

Table 3: Comparison result between Porter Stemmer and Vocabulary Based Stemmer (VBS)

	Porter Stemmer			Vocabulary Based Stemmer (VBS)		
	Unrecognized Word		Special Words	Unrecognized Word		Special Words
No	Before	After	Handling	Before	After	Handling
1	3	37	9	3	3	28
2	11	48	18	11	11	48
3	0	8	5	0	0	10
4	3	21	8	3	3	22
5	0	11	6	0	0	10
6	3	21	9	3	3	18
7	9	45	23	9	9	48
8	2	19	20	2	2	35
9	8	63	24	8	8	59
10	0	10	4	0	0	9

4. Conclusion

The objective of this paper was to develop a stemming algorithm based on vocabulary for any kind of application which relies on valid words and semantic relations between the words. To develop VBS stemmer and test the reliable stem output, and to ensure this stemmer can be applied on any English text, we ran the entire dictionary through the stemmer and check the result one by one. VBS has 3 additional lists which are containing the irregular verbs, exceptional words and special words. The total English candidate words (words with suffix ending) in the dictionary are 78,051 words, which double checked by comparing the original word definition and the stemmed word definition to ensure the successful stem. Through this process, exceptional words and special words are singled out and added to their appropriate list.

Acknowledgement

This project is funded by MoHE under research code FRGS/1/2016/ICT02/UKM/02/14.

References

[1] Bacchin, M., N. Ferro, and M. Melucci. "The effectiveness of a graph-based algorithm for stemming," in *ICADL. Springer.*2002.

[2] Lovins, J.B., "Development of a stemming algorithm,"*MIT Information Processing Group, Electronic Systems Laboratory Cambridge*.1968.

[3] Porter, M.F., "An algorithm for suffix stripping,"*Program,14(3):*1980.pp. 130-137. https://doi.org/10.1108/eb046814.

[4] Dawson, J.L., "Suffix removal and word conflation,"*ALLC Bulletin, Michaelmas,*1974. pp. 33-46.

[5] Dattola, R.T., "FIRST: Flexible information retrieval system for text,"*Journal of the Association for Information Science and Technology,* 1979. 30(1):pp. 9-14. https://doi.org/10.1002/asi.4630300103.

[6] Porter, M.F., "Snowball: A language for stemming algorithms," 2001.

[7] Willett, P., "The Porter stemming algorithm: then and now," *Program,* 2006. 40(3):pp. 219-223. https://doi.org/10.1108/00330330610681295.

[8] Van Rijsbergen, C.J., S.E. Robertson, and M.F. Porter, "New models in probabilistic information retrieval,"*British Library Research and Development Department*. 1980

[9] Chris, D.P. "Another stemmer," in *ACM SIGIR Forum.* 1990.

[10] Kraaij, W. and R. Pohlmann, "Porter's stemming algorithm for Dutch. Informatiewetenschap," 1994: pp. 167-180.

[11] Idris, N. and S.S. Mustapha, "Stemming for term conflation in Malay texts," 2001.

[12] Orengo, V.M. and C. Huyck. "A stemming algorithm for the portuguese language," in *String Processing andInformation Retrieval, 2001. SPIRE 2001. Proceedings. Eighth International Symposium on IEEE.*2001 https://doi.org/10.1109/SPIRE.2001.989755.

[13] Ramanathan, A. and D.D. Rao. "A lightweight stemmer for Hindi," in *the Proceedings of EACL*. 2003.

[14] Taghva, K., R. Beckley, and M. Sadeh. "A stemming algorithm for the farsi language. in Information Technology: Coding and Computing,"*2005. ITCC 2005. International Conference onIEEE.* 2005.

[15] Savoy, J., "Searching strategies for the Bulgarian language,"*Information Retrieval,* 2007. 10(6):pp. 509-529. https://doi.org/10.1007/s10791-007-9033-9.

[16] Savoy, J., "Searching strategies for the Hungarian language,"*Information processing & management,* 2008. 44(1):pp. 310-324. https://doi.org/10.1016/j.ipm.2007.01.022.

[17] Sawalha, M. and E. Atwell." Comparative evaluation of arabic language morphological analysers and stemmers," in *Proceedings of COLING 2008 22nd International Conference on Comptational Linguistics (Poster Volume)). 2008. Coling 2008 Organizing Committee*. 2008.

[18] Sharma, D., "Stemming algorithms: A comparative study and their analysis,"*International Journal of AppliedInformation Systems,* 2012. 4(3): pp. 7-12. https://doi.org/10.5120/ijais12-450655.

[19] Oard, D.W., G.-A. Levow, and C.I. Cabezas. "CLEF experiments at Maryland: Statistical stemming and backoff translation," in *Workshop of the Cross-Language Evaluation Forum for European Languages*. *Springer*.2000.

[20] Bacchin, M., N. Ferro, and M. Melucci, "A probabilistic model for stemmer generation,"*Information Processing &Management,* 2005. 41(1):pp. 121-137. https://doi.org/10.1016/j.ipm.2004.04.006.

[21] Majumder, P., et al., "YASS: Yet another suffix stripper,"*ACM transactions on information systems (TOIS),* 2007.25(4): pp. 18.

[22] Paik, J.H., D. Pal, and S.K. Parui. "A novel corpus-based stemming algorithm using co-occurrence statistics," in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. ACM*.2011. https://doi.org/10.1145/2009916.2010031.

[23] Miller, G. and C. Fellbaum, "Wordnet: An electronic lexical database,"*MIT Press Cambridge*.1998.