Rule Based Urdu Stemmer

Rohit Kansal Vishal Goyal G. S. Lehal
Department of Computer Science Punjabi University, Patiala
Assistant Professor, Department of Computer Science, Punjabi University Patiala
Professor, Department of Computer Science, Punjabi University Patiala

rohitkansal87@yahoo.co.in vishal.pup@gmail.com gslehal@yahoo.com

Abstract

This paper presents Rule based Urdu Stemmer. In this technique rules are applied to remove suffix and prefix from the inflected words. Urdu is well spoken language all over the world but less work has been done on Urdu stemming. Stemmer helps us to find the root of the inflected word. Various possibilities of inflected words like وں (vao+noon-gunna), ے (badi-ye), یاں (choti-ye+alif+noon-gunna) etc. have been identified and appropriate rules have been developed for them.

Keywords-Urdu Stemmer, Stemmer, Urdu,Rules

Proceedings of COLING 2012: Demonstration Papers, pages 267–276,
COLING 2012, Mumbai, December 2012.

1 Introduction

Stemming is the process in which inflected words are reduced to find stem or root. There are various inflected words that can be reduced to stem.

e.g. In English language :

1) Act can have inflected words like actor, acted, acting etc.

2) Words like fishing, fished and fisher can be reduced to root word fish.

Similarly in Urdu various possibilities have been identified and rules have been developed appropriate

	Inflected Word	Root Word
1.	لڑکیاں (laṛkīāṃ)	لڑکی (laṛkī)
2.	بستیاں (bastīāṃ)	بستی (bastī)
3.	گاڑیاں (gāṛīāṃ)	گاڑی (gāṛī
4.	کتابیں (kitābēṃ)	کتاب (kitāb)
5.	میلے (mēlē)	میلا (mēlā)

Table 1 Examples of Urdu Stemmer

1.1 Approaches

Stemming algorithms are classified under three categories- Rule Based, Statistical and Hybrid.

1) Rule Based approach - This approach applies a set of transformation rules to inflected words in order to cut prefixes or suffixes.

E.g. if the word ends in 'ed', remove the 'ed'.

2) Statistical approach - The major drawback of Rule Based approach is that it is dependent on database. Statistical algorithms overcome this problem by finding distributions of root elements in a database. There is no need to maintain the database.

3) Hybrid approach - It is combination of both Affix removal and Statistical approach.

Stemming is useful in Natural Language Processing problems like search engine, word processing problems and information retrieval. In this stemmer we have applied Rule Based Approach in which we apply rules on various possibilities of inflected words to remove suffixes

or prefixes. In Urdu, the only stemmer available to us is Assas-Band developed by NUCES, Pakistan which maintains an Affix Exception List and works according to the algorithm to remove inflections.

2 Background and Related Work

The only Stemmer available to us in Urdu is Assas-Band developed by NUCES, Pakistan which maintains an Affix Exception List and works according to the algorithm to remove inflections. It has been developed by Qurat-ul-Ain-Akram et al. (2009) using Rule based approach. Urdu word is composed of sequence of prefix, stem and postfix. A word can be divided into prefix-stem-postfix. First the prefix is removed from the word which returns stem-postfix sequence. Then postfix is removed and stem is extracted. This system gives an accuracy of 91.2 %.This system worked as a base paper for our system. It gave an idea that how Urdu words should be handled and what are the challenges faced in handling them. We have also used Rule Based Approach but it is different from Assas-Band.

In 1968 Julie Beth Lovins developed the first English Stemmer. Then Martin Porter developed Porter Stemming Algorithm which is most widely used technique for stemming in English. Other work related to Indian Languages are like Pratik kumar popat et al.2010 developed Stemmer for Gujarati using Hybrid Approach. In this system optimal split position is obtained by taking all the possible splits of the word and selecting the split position which occur maximum. It gives an accuracy of 67.8 %. Dinesh Kumar et al.2011 developed a Stemmer for Punjabi using Brute Force Technique. It employs a look up table which contains relation between root forms and inflected forms. To stem a word, table is queried to find a matching inflection. If a matching inflection is found associated root word is returned. It achieves accuracy of 81.27 %.Sandeep Sarkar et al.2008 developed Rule Based Stemmer for Bengali which achieve an accuracy of 89 %. Ananthakrishnan Ramanathan et al. developed a lightweight stemmer for Hindi using suffix removal method. Suffix removal does not require a look up table. It achieves an accuracy of 88 %.Vishal Gupta et al.2011 developed stemmer for nouns and proper names for Punjabi language using Rule based approach. Various possibilities of suffixes have been identified and various rules have been generated. The efficiency of this system is 87.37 %.

3 Urdu Stemmer

An attempt has been made to develop Urdu Stemmer using Rule Based Approach in which we have developed rules to remove various prefixes and suffixes. We have designed Rule Based Approach Urdu Stemmer which helps us to find stem of various inflected words. For this we have developed a graphical user interface in which we can enter the input directly or we can also browse files. Database has been maintained of root words along with their frequencies. The collection of 101,483 unique words has been done. The flowchart of Urdu Stemmer is given below which explains the system step by step in detail.

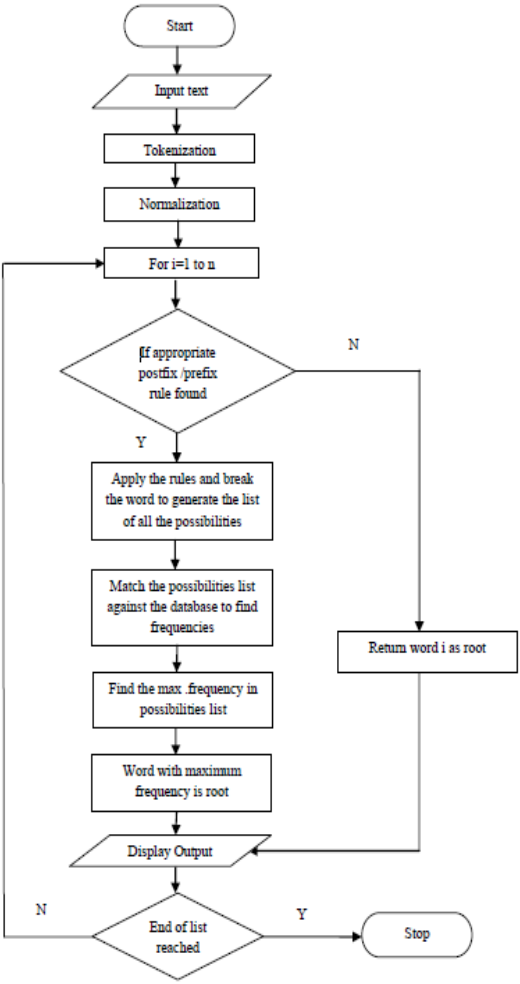

Figure 1 Flowchart of Urdu Stemmer

3.1 Algorithm

The algorithm of Urdu stemmer is explained in detail below:-

i) Tokenization and Normalization- In tokenization process the input text is tokenized word by word by using delimiter as space. In normalization special characters like ?,',",@ etc are eliminated.

ii) Postfix/Prefix Rules- After the normalization process postfix/prefix rules are applied on the word. If appropriate rules are found that can be applied then break the word and generate the list of various possibilities of the word. In some cases if appropriate rules are not found then system returns the same word as root word. The possibilities list is matched against the database to find frequencies. Then the frequencies are compared and word corresponding to the greatest frequency is returned as root. The word corresponding to the greatest frequency is returned as root because the word that occurs most frequently has the highest probability of being the root.
A corpus of 11.56 million words is used and 1,01,483 words are extracted from the corpus as unique words. These words are stored in the database along with their frequencies. The frequency of a word means how many times it repeats in the corpus.

Some of the postfix rules applied are-:

Rule 1- If word ends with وں (vao+noon-gunna) then remove وں (vao+noon-gunna) from end.

For example- رنگوں - رنگ
(raṅgōṃ) (raṅg)

Rule 2- If word ends with ے (badi-ye) then remove ے (badi-ye) from end and replace with ا (alif) .

For example- میلے - میلا
(mēlē) (mēlā)

Rule 3- If word ends with یوں (choti-ye +vao+noon-gunna) then remove یوں (choti-ye+vao+noon-gunna) from end and replace with ی (choti-ye).

For example- کویوں - کوی
(kaviyōṃ) (kavī)

Rule 4- If word ends with ؤں (vao- hamza+noon-gunna) then remove ؤں (vao- hamza+noon-gunna) from end.

For example- چاچاؤں - چاچا
(cācāōṃ) (cācā)

Rule 5- If word ends with یاں (choti- ye+alif+noon-gunna) then remove یاں (choti-ye+alif+noon-gunna) from end and replace with ی (choti-ye).

For example- کوٹیاں - کوٹی
(kōṭīyāṃ) (kōṭī)

Rule 6- If word ends with یں (choti- ye+noon-gunna) then remove یں (choti-ye + noon-gunna) from end.

For example- ڈھالیں - ڈھال
(ḍhālēṃ) (ḍhāl)

Rule 7- If word ends with ئیں (hamza + choti-ye+noon-gunna) then remove ئیں (hamza+choti-ye+noon-gunna) from end.

For example- مالائیں - مالا
(mālāēṃ) (mālā)

The rules above are some of the rules that are used in the Urdu stemmer. Similarly there are other postfix rules that can be applied which helps to find root in the system.

iii)Prefix Rules- Some of the prefix rules are applied to find the root word are given below:-

Rule 1- If word starts with بد (bay+daal) then remove بد (bay+daal) from beginning.

For example- بدصورت - صورت
(badsūrat) (sūrat)

Rule 2- If word starts with بے (bay+badi- ye)then remove بے (bay+badi-ye) from beginning

For example- بکدر - کدر
(bēkdar) (kadar)
So there are 32 postfix and prefix rules in total that we have used to develop this system.

4 Results and Discussion
We have tested this system on different Urdu news documents of 20,583 words to evaluate the performance of this system. The accuracy of this system is 85.14%. The news document consists of sports, national, international news. We have tried to cover different domains in order to find different types of inflected words. Test set 1 covers sports and business news. Test set 2 covers articles, short stories etc. Test set 3 covers news relating to health and science.

Test Set no.	Area covered	No. of Words
Test Set 1	Sports, Business news	7261
Test Set 2	Articles, Short stories	6239
Test Set 3	Health, Scientific news	7083

Table 2 Different test cases

Following evaluation metrics are used to calculate the accuracy.

Recall (R) = Correct answers given by system / Total possible correct answers

Precision (P) = Correct answers / Answers produced

F-Measure= $(\beta^2 + 1)$ PR / β^2 R + P

β is the weighting between precision and recall typically β=1. F-measure is called F1-Measure.

F1Measure= 2PR / (P+R).

Test set no.	Recall	Precision	F1-Measure
1.	90.90%	81.18%	86.11%
2.	88.39%	80.35%	84.17%
3.	89.43%	81.30%	85.15%

Table 3 Accuracy of different test cases

The overall accuracy of the system is 85.15%.The overall performance of the system is good. In test cases we have observed that some rules are more used than other rules. Rule 1 and Rule 2 cover most of the inflected words. So these rules are applied more than other rules. Errors are due to dictionary error or syntax error. Dictionary error means word is not present in the database. When we apply rules and find the various possibilities but these possibilities may not be present in the database. If appropriate rule is not found but the word is inflected it can also give rise to error. The probability of dictionary error is very less because we have extracted unique words from corpus of 11.53 million words. We assume that such a large corpus cover most of the inflected words. The error is mainly due to syntax error. There is no standardization in Urdu which means there is more than one way of writing a particular word. Although we have tried to cover all the possibilities of writing a word but error may occur. Absence of Airaabs in most of the Urdu text increases the error rate. When Airaabs are not present in Urdu text, it becomes difficult to understand the word. The different rules give different accuracy because some rules are more frequently used more than other rules.

The rules that are more frequently used are shown below and with their accuracy which helps to find which rule occur most. The rule that occur more frequently show that the inflection corresponding to that particular word occur most.

Urdu Stemmer Rules	Accuracy percentage of correct words
Rule 1 وں (vao+noon-gunna)	95.41%
Rule 2 ے (badi-ye)	94.74%
Rule 3 یوں (choti-ye+vao+ noon-gunna)	87.21%
Rule 4 ؤں (vao-hamza+ noon-gunna)	86.39%
Rule 5 یاں (choti-ye+alif+ noon-gunna)	84.11%
Rule 6 یں (choti-ye+ noon-gunna)	87.53%
Rule 7 ئیں (hamza +choti-ye+ noon-gunna)	85.41%

Table 4 Accuracy of mostly common applied rules

Conclusion and Future Work

In this paper Urdu stemmer has been discussed using Rule Based Approach which removes suffixes and prefixes from the inflected word. Various possibilities like وں (vao+noon-gunna), ے (badi-ye), یاں (choti-ye+alif+noon-gunna) etc. have been identified and appropriate rules have been developed to remove inflections and find the root. The data collection is the main problem because text data in Urdu available to us is rare. The limitation can be handled by increasing the database in future to achieve more accurate results. Error can also occur due to spelling variations because there is no particular way of writing a word. There can be more than one way of writing a particular word in Urdu. Although we have tried to put all the possibilities of writing a word but still error may occur. Statistical approach can be applied to Urdu Stemmer in future.

References

[1] Qurat-ul-Ain-Akram, Asma Naseer, Sarmad Hussain.(2009). *Assas –Band, an Affix-Exception list based Urdu Stemmer* In the Proceedings of the 7th Workshop on Asian Language Resources, ACL-IJCNLP , Suntec, Singapore, pp. 40–47.

[2] Dinesh Kumar, Prince Rana.(2011).*Stemming of Punjabi Words by using Brute Force Technique* In International Journal of Engineering Science and Technology (IJEST) Vol. 3 No. 2 , pp. 1351-1357.
[3] Sandipan Sarkar, Sivaji Bandyopadhyay.(2008). *Design of a Rule-based Stemmer for Natural Language Text in Bengali*, In the Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages, Hyderabad, India, Asian Federation of Natural Language Processing, pp. 65–72.
[4] Vishal Gupta, Gurpreet Singh Lehal.(2011). *Punjabi Language Stemmer for nouns and proper name*, In the Proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing (WSSANLP), IJCNLP , Chiang Mai, Thailand, pp. 35–39.
[5] Katik Suba, Dipti Jiandani, Pushpak Bhattacharyya.(2011).*Hybrid inflectional Stemmer and Rule-based Derivational Stemmer for Gujarati*, In the Proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing (WSSANLP), IJCNLP , Chiang Mai, Thailand, November 8, 2011, pp. 1–8.
[6] Pratikkumar Patel, Kashyap Popat.(2010). *Hybrid Stemmer for Gujarati"* In the Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing (WSSANLP), the 23rd International Conference on Computational Linguistics (COLING), Beijing, pp. 51–55.
[7] Ananthakrishnan Ramanathan, Durgesh D Rao *A Lightweight Stemmer For Hindi*, National Centre for Software Technology, In the Workshop on Computational Linguistics for South-Asian Languages, EACL. pp. 42-48.
[8] M.F. Porter, (1980). *An algorithm for suffix stripping,* Program, 14(3) pp. 130-137.
[9]http://en.wikipedia.org/wiki/Stemming_algorithms Accessed on November 2011
[10] Kashif Riaz.(2007). *Challenges in Urdu Stemming* (A Progressive Report In BCS IRSG Symposium: Future Directions in Information Access (FDIA 2007). http://citeseerx.ist.psu.edu/viewdoc/download/10.1.1.102.3051.pdf