A Hybrid Stemmer of Punjabi Shahmukhi Script

Abdul Mateen†, M. Kamran Malik†, Zubair Nawaz †, H. M. Danish†, M. Hassan Siddiqui †, Qaiser Abbas††

College of Information Technology, University of the Punjab, Lahore, Pakistan †
Department of Computer Science & IT, University of Sargodha, Sargodha, Pakistan††

Summary
Stemming is a heuristic process to chop off end part of words and sometimes adding additional letters at the end of words to get the basic meaningful forms of surface words. The basic goal of stemming is to reduce inflectional forms of words to root words using multiple techniques. In this paper, hybrid approaches are used for stemming Punjabi words. There has not been any stemmer reported for Punjabi شاہ مکھی (Shahmukhi) script. We used database lookup approach and rule based stemming for Punjabi Stemmer. Our dataset consists of 2.5 million tokens which were divided into three parts of 1500000, 500000 and 500000 tokens and used for training, development and testing purpose respectively. We got 86.01% accuracy while tested our stemmer over above specified dataset by using 63 rules.

Keywords:
Rule based stemmer, morphology, lookup approach, root words, hybrid stemmer, affixes and normalization.

1. Introduction

Stemming is a process to strip off root words and connected sub words from surface words. Root words are called stems and connected sub words are known as affixes. For example, in English "making and makes" both will be stemmed to root word "make". Here stem is "make" and affixes are "ing" and "s", respectively. Similarly, in Punjabi, word کڑیاں Kuryan (Girls) would be stemmed to کڑی kuri (Girl). More examples are given in Table 1.

Table 1: Examples of Punjabi شاہ مکھی (Shahmukhi) Stemmer

Sr.	Inflected words	Stem / Root
1	کڑیاں *Kuryan* (Girls)	کڑی *Kuri* (Girl)
2	منڈے *Munday* (Boys)	منڈا *Munda* (Boy)
3	منتاں *Mintaan* (Appeals)	منت *Minatt* (Appeal)
4	کتاباں *Kitaaban* (Books)	کتاب *Kitaab* (Book)

Punjabi is a language, which is widely spoken in Pakistan and India. Almost 100 million people have Punjabi as their native language in Pakistan, India, Canada and other European countries [16]. It has two types of scripts, one is گرمکھی(Gurumukhi) which is usually written in Hindi script and is spoken in Indian Punjab. Other script is شاہ مکھی(Shahmukhi) script which is written in Arabic script and follows Right to Left (RTL) language styles. Sufficient work has been done on گرمکھی(Gurumukhi) regarding information retrieval and text processing.

شاہ مکھی(Shahmukhi) is usually used in Pakistani Punjab for communication. Existing work using this script is deficient and disappointing.

Urdu also uses an RTL style and has the same script as شاہ مکھی (Shahmukhi), which supported us to make rules for our stemmer by using previous work done in Urdu like in [7].An important thing is that as Punjabi is morphologically rich language, due to which we have observed multiple scenarios in our work where stemming leads us towards wrong decisions like Over Stemming (stemming of a part which should not be stemmed) and Under Stemming (stemming of a part which should be stemmed). To handle these kinds of issues, we used additional techniques after stemming to make sure that the under and the over stemming are done properly. Table 2 shows some under and over stemming examples as follows. In this paper, the Punjabi stemmer is proposed for شاہ مکھی (Shahmukhi) script by using the hybrid approaches.

Table 2: Examples of false stemming

Sr.	Word	Stem/ Root	Stemming type	Correct Stem
1	کارکن kaar-kun (Worker)	کارک Kaark	Over stemming	کارکن Kaar-kun
2	پینڈواں Pendu-waan (villegers)	پینڈو Painduo	Under stemming	پینڈو Paindu
3	غدار ghaddar (Traitor)	غ Gayin	Over stemming	غدار Gaddaar

2. Background & Related Work

Lovins [1] discussed first about rule based stemmer for English. In this stemmer, almost 260 rules were used for stemming English words. One of the famous work done is by Porter [2] which is known as the rule based stemmer for English. He actually enhanced and simplified the work done in [1] and defined just 60 rules which are now famous due to versatility and usage. He proposed a rule based stemmer, in which series of rules were defined and words were categorized for stemming into classes and

Manuscript received August 5, 2017
Manuscript revised August 20, 2017

each class of words had specific rules. Porter's work was the best work in English stemming and it helped to draw conclusions about stemming on rule based approaches as it used just 5 steps for complete stemming.

A considerable work is also done on languages of Asian region. There are different stemmers for the RTL languages like Urdu, Arabic and Persian. As Punjabi is also RTL so the work done in above Asian region languages is helpful for Punjabi شاہ مکھی (Shahmukhi) Script stemmer. Urdu is quite similar to Punjabi and both use the Arabic script for writing. In [3] Husain used unsupervised approach for the Urdu stemming. He has done his work using N-grams and using two different techniques for stemming purpose. His algorithm is simply a 5-step algorithm, which first uses N-gram over Urdu corpus and splits suffix and stems. Then sort stems and suffix based on frequency and makes rules based on above 3 steps. One important point is that all those suffixes, which have count less than specified threshold are discarded and not included while making rules. Then two approaches were used, either to go for frequency-based approach or for length-based stripping off. Frequency based approach produced accuracy about 84.27% and the length-based approach gave 79.63% maximum accuracy for Urdu.

Gupta et al., suggested rule based stemmer for Urdu. They achieved accuracy up to 86.5% using different forms and variations of rules. Interesting thing in their suggested rules was that they categorized words into main categories like verb, adjective, noun etc., and provided surprising results based on category specified rules. Further, there were repetitive experiments performed to make sure that the extracted suffixes and stems were meaningful and logical. [4]

The work in [5] was a refined and updated version of [6]. Here newspaper corpus was used for stemming purpose. Database of suffixes was used to improve results and suffixes were categorized into 75 possible items. An efficient algorithm was provided which not only stemmed properly but also resulted 90% precision. This idea enhanced the accuracy, as this was the combination of algorithmic approach, use of a database and the rules for stemming.

Akram et al. [7] proposed an Assas-Band stemmer, which not only stem words but also improve the efficiency by maintaining proper classes and categories. For example, it maintained the distinction between feminine and masculine while stemming, which was a plus point for this rule-based stemmer. The technique used was simple, as a word is composed of prefix, stem and postfix so in first attempt, prefix was removed then postfix was removed and finally, the stem was processed further for exception handling. It also included rules list, which was used to remove, prefixes and postfixes from the words to get stems. After applying the techniques discussed, additional changes were also required in special cases to add extra characters to make the stems proper and meaningful words. In this phase, extra characters were added to enhance the efficiency of the overall process. In the last stage, manual corrections were applied to check whether proper classes' distinction has been maintained like masculine and feminine class or lookup approach has been used to verify any missing entity in the Prefix and the Postfix list, which were used for the initial processing. Overall maximum accuracy achieved was 91.18 percent.

A hybrid approach was used by Thapar in [8] for the Punjabi Stemmer of گرمکھی(Gurumukhi) script which is written in Hindi. Whenever a word was provided as input, it was searched in a database of root words, which had been already created by collecting the data. If a word found in the database, then it was not stemmed and displayed as a stem. If it was not found in the database, then proper stemming techniques were applied to find out root word. Similar kind of work has been done by Joshi & Garg in [9]. This paper included 3 techniques for stemming Punjabi words of گرمکھی(Gurumukhi) script. These 3 techniques included brute force approach, rule based approach and synset approach. Words were searched in tables if they found then it gave 100 percent accuracy. If not, then proper stemming was performed and after that over-stemming and under-stemming were properly handled in this stemmer. Further, to handle such suffixes, reduction or substitution techniques were used, which gave root words and then these words were verified by the lookup approach finally.

K Raiz in [10] highlighted challenges faced in Urdu stemming. Urdu being a rich language uses words from many other languages like Persian, Arabic, etc. Urdu has the Arabic and Persian orthography and also shares the grammar rules. Raiz explored the rule-based approach further with the following attributes like nouns, adjectives, pronouns, verbs, numbers, date & time, Persian loan words, Arabic loan words, gender & number agreement, etc. Other challenges that were described in his work by K Raiz, are the engineering issues. Out of 569 words, 211 were stemmed and out of these 211 words, 32 words were stemmed incorrectly. V. Gupta & G. S. Lehal worked on keyword extraction for Punjabi in [6]. According to them, this can be achieved in different phases like removing the stop words, identification of Punjabi nouns and noun stemming, calculating term frequency and inverse sentence frequency etc. Bundle of experiments were performed over 50 Punjabi documents for text extraction. The result of those experiments found precision on 80.4%, recall on 90.6% and F-Score on 85.2%. There was certain percentage of errors like 14.8% which was due to dictionary mistakes, absence of certain Punjabi nouns, syntax mistake in input text and violation of rules for certain noun stemming.

Stemmer1 for Quran (the book of God) used two type of stemming algorithms: prefix stemming and suffix stemming. All the stop words were excluded from the stemming process and this was achieved by using Quranic lexicon prepared by Fuad 'Abd Al Baqi [12]. The accuracy of stemmer was 99.6% for the prefix stemming and 97% for the suffix approach. Inaccuracies detected in the results were because of incorrect transliteration of lexical items [13].

Brute force approach as termed as exhaustive search was used to build the Punjabi stemmer [14]. In this approach a lookup table was used which possessed the relationship between the root word and the inflected word. If the word was not found in lookup table, then suffix stemming or suffix-striping algorithm came into the role. Accuracy of this stemmer was highly dependent on the number of words presented in the look up table. Larger the lookup table, better were the results.

V. Gupta & G. S. Lehal proposed the stemming for the Punjabi nouns and proper names. According to them, if stem words are present in the dictionary then it is noun else it is a proper name. Experiments were performed on the corpus of Punjabi and found 87.37% efficiency. They used rule base approach and created different orthographic rules but overall, error percentages were 9.78% due to rule violation, 2.4% due to dictionary mistakes and 0.45% due to spelling mistakes. [15]

Punjabi is similar to Urdu and in our stemmer, we used hybrid approaches which are discussed in the next section.

3. Proposed stemmer

We have proposed a hybrid stemmer for Punjabi شاہ مکھی (Shahmukhi) script. We have used a rule based and a lookup based approach for this stemmer.

The workflow is described in following algorithm and the basic flow of the work is given in the form of flowchart in Fig 1.

ALGORITHM 1. Hybrid Punjabi **شاہ مکھی (**Shahmukhi**)** Stemmer

Input: Set of Punjabi words.
Output: Properly stemmed roots.

1. Read input
2. Tokenize on space basis
3. Pre-Process/ Normalize (this includes removal of special characters and organizing the words properly)
4. *For* each word W_i in Tokens
   If W_i belongs to *Exception List* or *Lookup Table*
   Output *stem* against W_i and continue to step4;
   If wordLength (W_i) <=3
   Apply rule 4 given in Table 11 and move to step 4
   Else
   Remove prefix/postfix;
   Apply rules on W_i, output processed *stem*;
   If output word W_o belongs to dictionary, then add it to *Lookup Table* against W_i;
   Move to step 4;
5. End

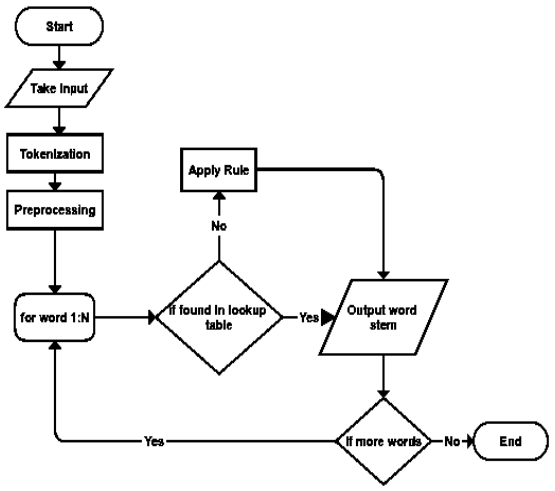

Fig 1: Flow chart of Punjabi Stemmer.

3.1 Details of Algorithm

The details of algorithm include different modules of our stemmer. All of these modules are sequential. Each module is described precisely as follows.

3.1.1 Tokenization

Input words are split based on space as a delimiter. After this phase, we have an array of words known as tokens, which is used further for stemming process.

3.1.2 Normalization

In normalization phase special characters @, #, *, &, etc., are removed so that better results could be achieved just in handling Punjabi text 2 and characters. In this phase, English and other languages text and words are removed.

3.1.3 Iterative Approach

Iterative approach is used to process the whole input dataset. All words are processed sequentially and during each iteration, hybrid3 approaches are used to get stem of the under process word.

[1] It was implemented using Delphi that is an object-oriented language and this stemmer was developed for windows.

[2] Punjabi is a language composed of words of Arabic Script. It is Right to Left (RTL) language and it contains mixture of Urdu, Arabic and Punjabi text words.

[3] Hybrid approach includes lookup base approach followed by a rule base approach.

3.1.4 Length Based Decision

If length of a word is less than 4 characters and it is already in root form, then there is no need to process that word. Examples are provided in Table 3.

Table 3: Length based decision

Sr.	Word	Stem / Root
1	مجھ Majj (Buffalo)	Majjمجھ
2	ماں Maan (Mother)	Maanماں
3	چھت Chatt (Roof)	Chattچھت

3.1.5 Exception List

Exception List includes all those words on which we can't apply any rule. These words should be included in output file as it is. These all words are maintained in a database list known as Exception List. All kinds of nouns and proper nouns can also be the part of this Exception List to produce accurate results. Table 4 and Appendix B show examples of such words.

Table 4: Exception List based decision

Sr.	Word	Stem / Root
1	ہاتھی Haathi (Elephant)	Haathiہاتھی
2	پاکستان Pakistan	Pakistanپاکستان
3	کھڈاری Khdaari (Player)	Khadaariکھڈاری

3.1.6 Prefix/Postfix Rule

We have two lists, which have all prefixes and postfixes. Word is checked against any of the prefix or postfix. If any prefix or postfix is found, it is removed. Table 5 and 6 show the lists of prefixes and postfixes, while the Table 7 presents the example of prefix and postfix rule.

Table 5: List of Prefixes.

با Beh+Alaf	نو Noon+Waw	بد Beh+Daal	ما Meem+Alaf	تو Teh+Waw	کم Kaaf+Meem
بے Beh+Bri-Yeh	بالا Beh+Alaf+Laam+Alaf	نا Noon+Alaf	مہا Meem+Heh+Alaf	لا Laam+Alaf	ان Alaf+Noon

Table 6: List of Postfixes.

باز Beh+Alaf+Zeh	دار Daal+Alaf+Reh	فروش Feh+Reh+Waw+Sheen	گاہی Gaaf+Alaf+Heh+Choti-Yeh
گار Gaaf+Alaf+Reh	ناک Noon+Alaf+Kaaf	خور Kheh+Waw+Reh	بین Beh +Choti-Yeh+Noon

Table 7: Prefix/Postfix based decisions.

Sr.	Word	Stem/ root	Rule Type
1	بااصول Baa-Usool (Disciplined)	اصول Usool	Prefix
2	دغا باز Dagha Baz (Treacherous)	دغا Dagha	Postfix

3.1.7 Lookup Table

A majority of words are stored in Lookup Table, a database table. These words are stored along with their stems. A word is first searched in Lookup Table. If a word belongs to the table, then its stem is returned and this shows 100 percent accuracy for that specific word. It is a little bit different from the Exception List. In Exception List words are added before processing and in dictionary words are added along their stems after complete processing. This Lookup Table along with the Exception List is verified manually and it enhances our accuracy and efficiency.

3.1.8 Rules Based Approach

In this phase, it is considered that a given word is totally unseen, as it does not belong to the Lookup Table or the Exception List. To handle such words, we have defined a set of 63 rules, which stems these words properly.

3.2 Algorithmic Rule Based Approach

Few important rules are mentioned here for all words of length > 3, which are also not in the Lookup Table or the Exception List.

3.2.1 Rule 01

If a word ends with اں (Alaf + Noon Gunna) then remove اں from the end of the word. Examples are given in Table 8.

Table 8: An example set for Rule 01.

Sr.	Word	Stem/ root
1	ونگاں Wangaan (Bangles)	ونگ Wangg
2	چھتاں chatatan (Roofs)	چھت Chatt

3.2.2 Rule 02

If a word ends with ے (Bari yeh) then remove ے from the end of the word and insert ا (Alaf). Table 9 shows a few relevant examples of this rule.

Table 9: An example set for Rule 02.

Sr.	Word	Stem/ root
1	گانے Gaany (Songs)	گانا Gaana
2	چووے Choowy (Rats)	چووا Choowa

3.2.3 Rule 03

If a word ends with ین (Choti yeh+Noon) then remove ین from the end of the word. Examples are given in Table 10.

Table 10: An example set for Rule 03.

Sr.	Word	Stem/ root
1	شوقین Shoqeen (Fond of)	شوقShouq
2	رنگین Rangeen (Colorful)	رنگRang

3.2.4 Rule 04

If a word length is between 3 and 5 and the word ends with و (Waw) then remove و from the end of the word. Examples are given in the following Table 11. Based on the variation and different variant forms of the words, there is a variety of words in Punjabi. If words length is less than 4 then there is more than 98 percent chance that the word itself is root.

Table 11: Example set for Rule 04.

Sr.	Word	Stem/ root
1	ورک Kro (Do this)	رکkar
2	وھکےو Waikho (Look)	ھکےو Waikh
3	پجو Pajjou (Run)	چپ Pajj

All our rules cannot be presented here. Few important ones are discussed and the rest are depicted in appendices.

4. Experimental Evaluation

For our Punjabi stemmer, we collected dataset from online resource [11] of Punjabi. In this section, experimental details over specified dataset of Punjabi are prescribed.

4.1 Experimental Dataset

We have collected dataset of 2.5 million words which has 85152 unique words from online resources [11]. Our dataset is mixture of literature, politics, and science and poetry words. We divided data into three parts i.e. training data C1, development data C2 and testing data C3. Using C1 we have developed initial list of rules and then using C2 we improved our rules to get better results. Table 12 shows the number of tokens and unique words in each part.

Table 12: An overview of corpus

Corpus Number	Total words	Unique words
C1	1500000	35168
C2	500000	25073
C3	500000	24911
Total	2500000	85152

4.2 Minimum Word Length Rule

During experiments, we have observed that 98 percent words of length less than 4 need no stemming and these are already in base or root form. Few examples are given in Table 13.

Table 13: Example set for MinWordLength rule

Sr.	Word	Root / Stem
1	کر Karr (Do)	کر Karr
2	جا Jaa (Go)	جا Jaa
3	بول Bol (Speak)	بول Bol

4.3 Lookup Table Updating Rule

In our first experiment, we used the rule-based approach over our corpus C1. The results are not good and there is a verity of under and over stemming. An example set is given in Table 2. All those words, which are not stemmed properly then added into the Lookup Table with their correct stem and updated Exception List.

4.4 Rules Revision Policy.

While working over 1.5 million words of Punjabi corpus, we have observed that few rules are giving inaccurate results and instead of enhancing accuracy, these rules are decreasing efficiency of our stemmer. Updated rules are given in Appendix A.

4.5 Accuracy Calculations.

Accuracy of our stemmer is calculated through equation 1.

$$Accuracy\ (\%) = \frac{Correctly\ stemmed\ words}{Total\ tested\ words} * 100 \quad (1)$$

We have used C2 for Phase 01, Phase 02 and Phase 03 and C3 have been used for Phase 04 experimentation.

4.6 Stemming Process

Stemming process is divided into the following phases.

4.6.1 Phase 01

In Punjabi Stemmer, the rule-based approach has been used in Phase 01 of accuracy calculations over the development data C2. An applied algorithm given in section 3 has been tested on 25073 words and has got 8956 words stemmed properly with accuracy 35.71%. Details are provided in Table 14.

4.6.2 Phase 02

In second phase, we have manually put the correctly stemmed words into the Lookup Table and updated our rules based on the remaining un-stemmed words. We have learnt from experiments in this phase that few words are better not to stem up to the last level, for these kinds of words, better approach is to stem up to a reasonable extent. For example, بدمعاش (Badd-Muaash) would be stemmed to معاش(Muaash) instead of ش‌اع‌مردب (Badd-Muaash). In fact, few words should not be stemmed or stemmed to a reasonable extent. Examples are given in Appendix B.
For phase 02, the dataset C2 is used and we have applied both the approaches: rule based approach and the Lookup Table approach. For this phase, we have used 25073 unique words and we have got 17804 words correctly

stemmed with accuracy of 71.01%. Results are listed in Table 14.

4.6.3 Phase 03

In this phase, rules are modified and the Exception List is included and we put all those words in EXCEPTION, which occur in our corpus with a huge frequency and follow no rules. We populated our Lookup Table by correcting the under and over stemmed words manually. For phase 03, we have 26494 words in our Lookup Table help in improving the accuracy, which is clear in Table 14. Some of the rules are defined in Appendix A.

4.6.4 Phase 04

In this phase, dataset C3 is used. We have modified rules and updated Lookup Table with 30494 words. All these words are properly stemmed and we verified accuracy of our testing data through manual comparison.
We used 0.5 million words data set having 24911 unique words. From which our algorithm correctly stems 21426 words. Results are mentioned in Table 14 and Fig 2.

Table 14: Phase wise results.

Phase No.	Total words	Unique words	Correct stemmed	Accuracy %
01	500000	25073	8956	35.71
02	500000	25073	17804	71.01
03	500000	25073	20517	81.82
04	500000	24911	21426	86.01

5. Results & Discussions

We used the Lookup Table, Exception List and the repeatedly modified rules. During phase 02 and 03, we modified, eliminated and added new rules to improve the accuracy of the proposed stemmer. In the end, Lookup Table became so large to handle almost majority of unseen words. Currently, the Lookup Table consists of 30494 unique words, which can be increased to achieve more accuracy. Proposed stemmer produced 86.01% accuracy using both the rules and the Lookup Table. Without using Lookup Table, proposed stemmer produced 35.71% accuracy in phase 01 and 64.90% in phase 04.

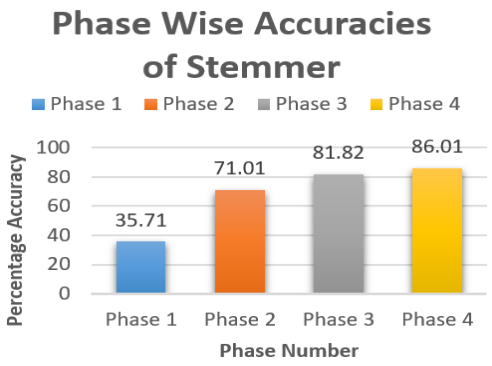

Fig 2: Stemming accuracy in all phases.

6. Limitations

Proposed stemmer uses rule based and lookup based approaches for stemming Punjabi words. As Urdu and Punjabi has a lot of common vocabulary, so, the proposed stemmer by default handles the Urdu data up to some extent. Proposed stemmer is efficient when it uses both rules and the Lookup Table. Rules are defined, which covers major part of Punjabi data but not all words.

7. Conclusion & Future Work

The proposed Punjabi stemmer for شاہ مکھی(Shahmukhi) script first uses the length based decision to output the proper stem. After taking this length-based decision, for words greater than 3 characters' length first it uses the lookup approach, which includes lookup from the Lookup Table and the Exception List. This stemmer is first such stemmer for شاہ مکھی (Shahmukhi) script and gives us promising results with 86.01 percent accuracy by using the prefix, postfix and infix removal methods. This accuracy can be improved by improving the Lookup Table. In future, the rules should be increased and the Lookup Table should be enlarged, this will give us more accuracy for our stemmer. Furthermore, statistical approaches can also be used to decide the best root for a given word.

References

[1] L. Julie, "Development of a stemming algorithm". Mechanical Translation and Computational, 1968.
[2] M. Porter, "An algorithm for suffix stripping". Program: Electronic Library and Information Systems, 14(3), 130-137. doi:10.1108/eb046814, 1980.
[3] Husain, M. Shahid, "An unsupervised approach to develop stemmer". International Journal on Natural Language Computing (IJNLC) 1.2 (2012): 15-23
[4] Gupta, Vaishali, Joshi, Nisheeth, & M. Iti, "Rule based stemmer in Urdu". In Computer and Communication Technology (ICCCT), 2013 4th International Conference on (pp. 129-132). IEEE, 2013.

[5] Puri, Rajeev, Bedi, R.P.S., & Goyal, Vishal, "Punjabi stemmer using punjabi wordnet database". Indian Journal of Science and Technology, 8(27), 2015.
[6] V. Gupta & G. S. Lehal, "Automatic keywords extraction for Punjabi language". International Journal of Computer Science Issues, 8(5), 2011.
[7] A. Qurat-ul-Ain, Naseer, Asma, & H. Sarmad, "Assas-Band, an affix-exception-list based Urdu stemmer". In Proceedings of the 7th workshop on Asian language resources (pp. 40-46). Association for Computational Linguistics, 2009.
[8] Thapar, Puneet, "A Hybrid Approach used to Stem Punjabi Words". 2014.
[9] G. Joshi, G. Kamal Deep, "Enhanced Version of Punjabi Stemmer using Synset". International Journal, 4(5), 2014.
[10] Riaz, Kashif. "Challenges in Urdu Stemming (A Progress Report)." In Proceedings of the 1st BCS IRSG conference on Future Directions in Information Access, pp. 4-4. British Computer Society, 2007.
[11] WICHAAR DOT COM***** Punjabi News and Comprehensive Punjabi Journal*****, Wichaar.com. Retrieved 14 July 2016, from http://www.wichaar.com,
[12] Abdul Baqi, M. F. "Al-Mu'jam al-Mufahras li Alfaz al-Qur'an al-Karim, 1987. [The dictionary of the phrases of the Glorious Quran]." Cairo: Dar al-Hadith.
[13] Thabet, Naglaa. "Stemming the Qur'an." In Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages, pp. 85-88. Association for Computational Linguistics, 2004.
[14] K. Dinesh & R. Prince, "Design and Development of a Stemmer for Punjabi". International Journal of Computer Applications, 11(12), 18-23, 2010. http://dx.doi.org/10.5120/1634-2196
[15] V. Gupta & G. S. Lehal. "Punjabi Language Stemmer for nouns and proper names." In the Proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing (WSSANLP), IJCNLP, Chiang Mai, Thailand, pp. 35-39. 2011.
[16] Christopher Shackle, "Punjabi language". Encyclopedia Britannica. Retrieved 14 July 2016, from https://www.britannica.com/topic/Punjabi-language, 2016.

Abdul Mateen received the B.S. in Software Engineering from Punjab University College of Information Technology, Lahore in 2015. Currently he is working as Software Developer at Nakisa Solutions Pakistan and perusing his Masters in Computer Science from Punjab University College of Information Technology, Lahore.
His research interests are Natural Language Processing and Machine Learning.
Muhammad Kamran Malik pursuing his PhD in Computer Science from Punjab University College of Information Technology, Lahore. He is serving as Assistant Professor at Punjab University College of Information Technology since September 2010.He received Silver Medal in MS Software Project Management. His research areas are Natural Language Processing, Machine Learning and Artificial intelligence.

Zubair Nawaz is serving as Assistant Professor at Punjab University College of Information Technology. He completed PhD in Computer Engineering from Delft University of Technology in 2011. His research areas are Data Science, Analysis of Algorithms, Advanced Algorithms and High Performance Computing.
Hafiz Muhammad Danish received B.S. degree in Computer Science from Punjab University College of Information Technology in 2015 and doing M.S. in Computer Science. During 2015 -2016, he has worked as a Software Engineer in Systems Limited, Pakistan. Now he is working as a lecturer in PUCIT since October 2015. His research areas are Natural Language Processing and Compiler Constructions.
Muhammad Hassan Siddiqui received the B.S. in Electrical with major in Electronics from University of Engineering and Technology, Lahore in 2013. Currently he is perusing his Masters in Computer Science from Punjab University College of Information Technology. His research interests are Natural Language Processing and Speech Synthesis.

Qaisar Abbas received his PhD (Computational Linguistics) from University of Konstanz, Germany in 2014. Now he is working as Assistant Professor at University of Sargodha. His research areas are Natural Language Processing/Computational Linguistics, Data & Text Mining / Information Retrieval

Appendix A: Few Rules for Punjabi Stemmer

Rule Description	Examples. Word Stem	
If a word ends with ان (Noon+Alaf) then remove ان from end of the word	اناج Jana	اج Jaa
If a word ends with راک (Kaaf+Alaf+Reh) then remove راک from end of the word	صالح راک Slah Kaar	صالح Slaah
If a word ends with اواں (Alaf+ Waw+Alaf+Noon -gunnah) then remove واں from end.	سکھاواں Sikhawan	سکھا Sikhaa
If a word ends with راگ (Gaaf+Alaf+Reh), remove راگ from end of the word.	راگدم Madad Gaar	دم Madad
If a word ends with کان (Noon + Alaf + Kaaf), remove کان from end of the word.	دردکان Dard Naak	درد Dard
If a word ends with روخ (Kheh + Waw + Reh), remove روخ from end of the word.	مرداً روخ Adam Khour	مرداً Adam
If a word ends with ی*ے (Choti-yeh+Any character +Bri-yeh) then remove ےfrom end of the	گھسیٹے Ghseety	گھسیٹ Ghseet

word.		
If a word has lenth >=6 and word ends with دیاں (Daal + Choti yeh + Alaf+Noon gunnah) then remove دیاں from end of the word.	لشکدیاں Lashkdyaan	لشک Lashak
If a word starts with خش (Kheh + Sheen) then remove خش from start of the word.	خٹ خش Khush khatt	خش Khush
If a word starts with مہ (Heh + Meem) then remove مہ from start of the word.	رمع مہ Hamm Umar	رمع Umar
If a word starts with ھدا (Alaf +Daal+ Heh) then remove ھدا from start of the word.	ادھ موم ھدا Adh moya	ادھ موم Moya
If a word starts with ان (Noon + Alaf) then remove ان from start of the word.	دارم ان Na muraad	دارم Muraad
If a word ends with نو (Waw + Noon) then remove نو from end of the word	چمکاوں Chamkawann	اکمچ Chamka
If a word ends with ن (Noon) then remove ن from end of the word.	کرن Karan	کر Karr
If a word ends with راد (Daal + Alaf + Rey) then remove روخ from end of the word.	اامارں راد Emaarn Daar	اامارں Emaan
If a word ends with وں (Waw + Noon gunnah) then remove وں from end of the word.	انالوں Naaloon	انال Naal
If a word ends with لن (Laam + Noon) then remove ن from end of the word.	بولن Bolann	لوب Bol
If a word has length >=5 ends with اوے (Alaf+Waw+Bri-yeh) then remove اوےfrom end of the word.	ڑلاوے Laraway	ڑل Larr
If a word ends with اندے (Alaf+Noon+Daal+Bri-yeh) then remove ےدنfrom end of the word.	منگواوےدن Mangwan-day	منگوا Mangwa
If a word has length>5 and ends with ںاوو (Waw+Waw+Alaf+Noon gunnah + Teh) then remove تا from end of the word.	جنگوجاں Jangjouwaan	جنگوج Jang-Ju

Appendix B: Exceptions

Rule Description	Example	Exception word
If a word ends with ی*ے (Choti-yeh+Any character +Bri-yeh) then remove ےfrom end of the word.	گھسیٹے Ghseety	سلیقے Saleeqy
If a word ends with اں (Alaf+Noon gunna) then remove اں from end of the word.	عورتاں Aourt-aan	دعاواں Duawaan
If a word length>=5 and word ends with انے (Alaf +Noon+Bri-yeh) then remove انے from end of the word.	دبانے Dabany	کمائے Kamaany
If a word length >=6 and word ends with اندے (Alaf+Noon+Daal+Bri-yeh) then remove ےندfrom end of the word.	منگوادے Mangwa day	پراندے Parandy
If a word has length>3 and word ends with ئے (Hamza Choti yeh +Bri yeh) then remove ئے from end of the word	سیکھائے Sikhaaye	ائے Aaye
If a word has length >=5 ends with اوے (Alaf+Waw+Bri-yeh) then remove اوےfrom end of the word.	لڑاوے Laraway	جاوے Jaawy
If a word ends with ین (Choti-yeh+Noon) then remove ا ین from end of the word.	شوقین Shoqeen	ائین Aaein
If a word starts with بد (Beh+Daal) then remove بد from start of the word.	بدصورت Badd-Sortt	بدتر Badd-Tarr
If a word starts with مہا (Meem+Heh+Alaf) then remove مہاfrom start of the word.	مہاراجا Maha-Raja	مہاتما Maha-Tama