International Journal of Electrical and Computer Engineering (IJECE)
Vol. 11, No. 3, June 2021, pp. 2307~2314
ISSN: 2088-8708, DOI: 10.11591/ijece.v11i3.pp2307-2314

Tigrigna language spellchecker and correction system for mobile phone devices

Atakilti Brhanu Kiros[1], Petros Ukbagergis Aray[2]
[1]Faculty of Computing Technology, Aksum University, Aksum, Ethiopia
[2]Department of Information Technology, Aksum Polly Technique College, Aksum, Ethiopia

Article Info

Article history:

Received Mar 9, 2020
Revised Oct 14, 2020
Accepted Oct 25, 2020

Keywords:

Android smart phones
Proto type model
Spelling checker
Spelling corrector
Tigrigna language

ABSTRACT

This paper presents on the implementation of spellchecker and corrector system in mobile phone devices, such as a smartphone for the low-resourced Tigrigna language. Designing and developing a spell checking for Tigrigna language is a challenging task. Tigrigna script has more than 32 base letters with seven vowels each. Every first letter has six suffixes. Word formation in Tigrigna depends mainly on root-and-pattern morphology and exhibits prefixes, suffixes, and infixes. A few project have been done on Tigrigna spellchecker on desktop application and the nature of Ethiopic characters. However, in this work we have proposed a systems modeling for Tigrigna language spellchecker, detecting and correction: A corpus of 430,379 Tigrigna words has been used. To indication the validity of the spellchecker and corrector model and algorithm designed, a prototype is developed. The experiment is tested and accuracy of the prototype for Tigrigna spellchecker and correction system for mobile phone devices achieved 92%. This experiment result shows clearly that the system model is efficient in spellchecking and correcting relevant suggested correct words and reduces the misspelled input words for writing Tigrigna words on mobile phone devices.

Corresponding Author:

Atakilti Brhanu Kiros
Faculty of Computing Technology
Aksum University
1010 Aksum University, Tigrai, Ethiopia
Email: ab.fruit@gmail.com

1. INTRODUCTION

The documents in many languages have been digitized and are available in different media especially on the web. Giant software vendors (e.g., Google and Microsoft) are also localizing their products to the native languages of their target customers. There is a need to develop computational solutions to the classic problems of computational linguistics for the respective languages. Spellchecking and correction is an application that identifies misspelled words, provides appropriate suggestion to misspelled word and ranks the suggested words for correction with the words that has high probability in the given list [1], which is checking ether they are rightly spelled or wrongly spelled and suggesting possible alternatives word [2]. Spellchecker and correction is a known and hot researched issue in natural language processing [3-6], and their solution can be used in many applications [7]. At present the computing technology is moving into our day to day life. The computing paradigm is shifting towards hand held devices. Currently, these android mobile devices are becoming widely used in the world even in our country [8]. Introducing texts to word processing tools in the mobile phone devices may have result in spell errors. Hence, various text processing software tools has spellcheckers. Integrating spellchecker into mobile phone devices increase the quality of

information and efficiency. The applications of these devices are mostly with foreign languages. If these devices can provide their services in the local languages Tigrigna, they will gain wide acceptance among the users and more application can be developed using the local language Tigrigna. Then, fast and error free spelling checker method is important thing for Tigrigna language writers on mobile phone. However, these tools are not available for the Tigrigna language. To improve the quality of life for the users specially by creating a mobile application that will help them communication effectively. Many Tigrigna language speakers uses mobile phones. Why those users cannot make their language part of the technology‘s language? In addition, the language can serve as an alternative text entry method for mobile phone like SMS.

Therefore, this study proposed a systems modeling for Tigrigna language spellchecker, detecting and correction. To indication validity of the spellchecker and corrector model and algorithm designed, a prototype is developed. The experiment is tested and accuracy of the Prototype achieved 92% based on ISO 9241 usability engineering standards. This experiment result shows clearly that the system model is efficient in checking and correcting relevant suggested correct words and reduces the misspelled input words for writing Tigrigna words on mobile phone devices.

2. ABOUT TIGRIGNA LANGUAGE

The language Tigrigna is one of the Ethio-Semetic languages which belongs to Afro-Asiatic super family that originated from the ancient Geez language, it is manly spoken in the East African countries of Tigrai national regional state of Ethiopia and Eritrea. Also, Tigrigna language is the communication and official language of Tigrai national regional state and Eritrea [9]. Therefore, there are more than six million Tigrigna language speakers worldwide [10].

“ትግርኛ ሓደ ካብ ሴማውያን ዓሌት ቋንቋ እንትከውን። ቋንቋ ትግርኛ ኣብ ኣባዛሉ ሀገረ ኤርትራን ኣብ ሰሜን ኢትዮጵያን ክልል ትግራይ ዝዝረብ ቋንቋ እንትኸውን። ፊደላት ቋንቋ ትግርኛን ኣምሓርኛን ካብ ኣደኦም ዝኾነ ቋንቋ ግእዝ ዝተወሰዱ እዩም።” is a sample Tigrigna text.

Let’s have a look here: Unlike the Latin language, the Tigrigna script has more than 32 base letters with seven vowels each. Every first letter has six suffixes. Word formation in Tigrigna depends mainly on root-and-pattern morphology and exhibits prefixes, suffixes, and infixes. Tigrigna is morphologically-rich in the way that grammatical relations and syntactic information are indicated at the word level. A few project have been done on Amharic and Tigrigna spelling checker on desktop application and the nature of Ethiopic characters has been clearly discussed.

3. RESEARCH METHOD

This has given us a clear idea on how to model the Tigrigna language spellchecker and correction system for mobile phone devices, design appropriate algorithms for Tigrigna spellchecker and correction system and instrument needed.

3.1. Process model (prototyping model)

This study uses prototyping process model for the development of the Prototype. The model is developed based on the current well-known requirements. This model allows the users to interact and experiment with a working model of the system called as prototyping mode [11-12]. The system lifecycle phases that are used in the development of prototyping model shows in Figure 1.

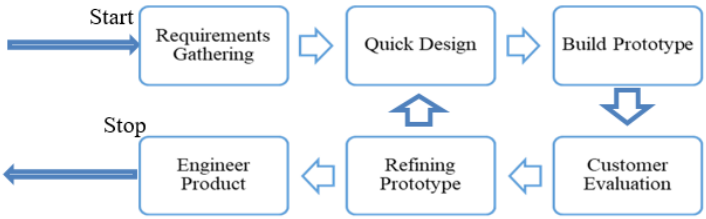

Figure 1. Prototyping model of system lifecycle

3.2. Data collections

To achieve a task of spellchecker and correction model, we collect Tigrigna word corpuses with 430,379 Tigrigna word, statistical information of Tigrigna words such as the frequency of occurrence of

words and word-length. Tigrigna words in a database in order to design the systems modeling for Tigrigna spellchecker and corrector system in mobile phone devices. Having a good collection of words and larger and large size dictionary in the corpus database helps to design and develop better spellchecker and correction model. Tigrigna word corpuses prepared from different sources that are from newspapers, magazines, books and from other Tigrigna written documents. Once the requirements of the spelling checker and correction completely done, the Tigrigna language user accepts the final prototype. In order to test this Tigrigna Spelling checker the proponent collect data from different user which they use to test the system. Finally by taking the sample size tested using 800 words by 20 evaluator each evaluator use 40 words. This can be done using Slovin formula [13].

$$n = \frac{N}{1+Ne^2}$$

Where: N = word size=430,379
e = margin of error (1%-5%)
n = sample size=800

3.3. Architecture of the system

In the processing of spellchecker and corrector modelling is divides into three different stages (Phases): The first stage, detecting of errors, during this level the lexical analyzer detecting the misspell word in the input string. The second, make suggestion for detecting misspelled word, during this also the system create a set of possible candidates as possible replacement for the inputted misspell word. And the last also, ranking suggestions and automatic corrections of errors, during which these candidates are sorted out from the most likely replacement to the smallest possible ones based on their associated error weight. Most techniques treat each stage as an isolated processing and performing them in sequences. Figure 2 shows the processing of Tigrigna language spellchecker and corrector system clearly for the readers in this respect.

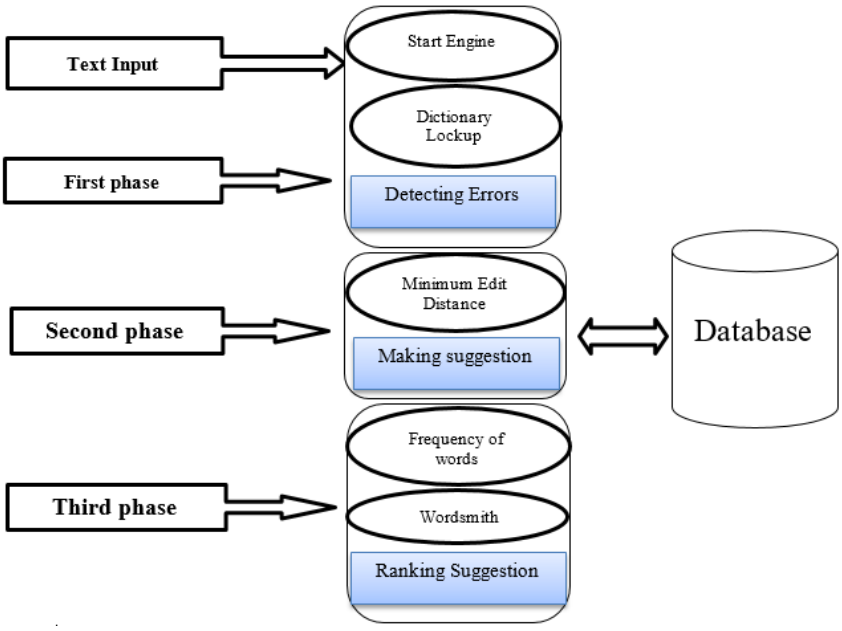

Figure 2. Tigrigna spelling checker system adapted from [14]

- First phase: *Start engine component*, is the first time gets one inserted word on the text input box. After these characters are received, initiate the word detecting error by comparing from the database. A dictionary lookup is one of the algorithm which helps for checking the data entered by users whether correct or misspell.
- Second phase: *Make suggestion*, that is Deletion, Insertion, and Deletion or Substitution required to change one word into the other. On this stage the detected error word make suggestion based on the rule of minimum edit distance.

– Third phase: *Word ranker*, by considering the frequency of each word, provides a rank to each words in the list of found words. Words with highest frequency will get highest rank and those with least frequency will get the least rank. Ranking policies are used to determine the word(s) to be suggested or corrected.

As mobile devices have screen size constraints, it is not possible to display large number of predicted words so that a user can select his/her desired word from the list. Thus, some trials are done to decide the convenient number of words to be displayed. As a result of these trials, it is found that displaying six (6) relevant words at a time will be much convenient. Hence, the Relevant Word suggestion Engine will display the top six or below ranked words as an output of the system.

3.4. Systems modeling for tigrigna spelling checker and correction

The two main functions provides by a spellcheckers are spelling error detecting and spelling error correcting. Error detection is to verify the validity of a word in the language while error correction is to suggest corrections for the misspelled word [15-16]. Figure 3 shows the Spellchecker and corrector system.

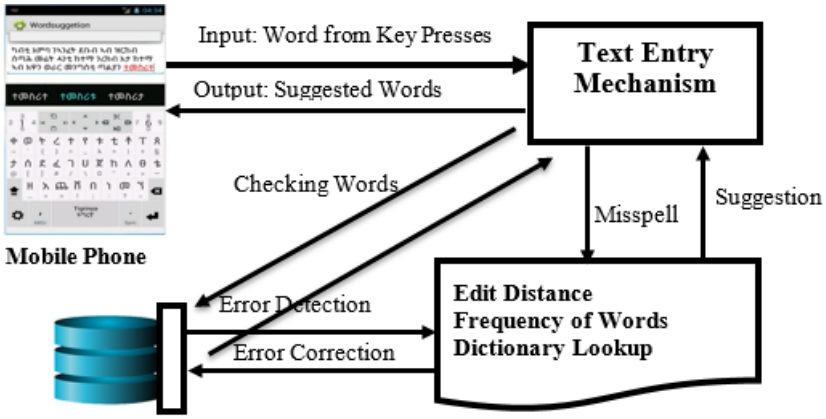

Figure 3. Tigrigna spelling checker system framework adopted from [17]

3.5. Techniques for correcting spelling error

3.5.1. Dictionary lookup

The objective of a Tigrigna spellchecker and correction is suggesting and completing the word a user is need to type a dictionary lookup is one of the algorithm which helps to check the data entered by users whether correct or misspell. A dictionary is a list of words that are assumed to be correct [18-19].

3.5.2. Edit distance

The minimum edit distance is the minimum number of operations (insertions, deletions and substitutions) required to transform one text string into another. In its original form, minimum edit distance algorithms require original form, minimum edit distance algorithms require m comparisons between misspelled string and the dictionary of m words. After comparison, the wordswith minimum edit distance are chosen as correct alternatives [19-20].

– Distance(“ደ/ፈ/አ”, “ደ/ፍ/አ”) => 1 (deletion)
– Distance(“ማ/እ/ት”,“ሰ/ማ/እ/ት”) => 1 (insertion)
– Distance(“ሀ/ሮ/ም”,“ጀ/ሮ/ም”) => 1 (substitution)

3.5.3. Levenshtein distance

Levenshtein distance (LD) is a measure of the similarity between two strings, the source string (s) and the target string (t). The distance is the number of deletions, insertions, or substitutions required to transform s into t [21].

3.5.4. Statistical spelling suggestion using frequency

The simplest form to minimize the data that is suggested by the algorithm dictionary lockup and edit distance is using frequency of words. Suggestion uses a fixed lexicon. Each of the words contained within it have a frequency score associated with relating to how often it is used in the language in general [22]. These

corpora are usually based on textual or written language although they are sometimes derived from spoken language. Table 1 shows the lists of thirty most frequently used distinct words.

The corpus are used to produce statistical information's of the word-length of each words and average word-length [22-24] of Tigrigna languages using wordsmith. Wordsmith tool, is used to find the statistical information like the average word-length from the corpus which are developed. This corpus is valuable in order to map ranking policy of the frequencies word or words [22]. The result of the Table 1 and Table 2 indicates the most frequently used word-length in Tigrigna language is 5-letters length. The analysis made to identifying the most frequently used Tigrigna word length and the length of each words in corpus helps to provide a clear idea on the frequency of the Tigrigna language.

Table 1. List of top 30 distinct words

No	Frequency	Words
1	203851	ኣብ
2	70351	ናይ
3	56687	ካብ
4	46018	እቲ
5	41928	ምስ
6	40221	ከም
7	38702	ድማ
8	29739	ናብ
9	28806	እዩ
10	23066	ከኣ
11	21459	ግን
12	21013	እዚ
13	20638	ሓደ
14	19855	ነቲ
15	18791	ትግራይ
16	18528	ዘሎ
17	16768	ግዜ
18	14954	ሰብ
19	14232	ኢዩ
20	13585	ዞባ
21	12812	ወይ
22	12699	ኮይኑ
23	12644	ድሕሪ
24	11995	ኣሎ
25	11972	ኣኖ
26	11805	ኣብቲ
27	11447	ምኳኑ
28	11328	ጥራይ
29	11066	ክሳብ
30	10817	ዓመት

Table 2. Word lengths on the tigrigna corpus

Data item	Total document
Distinct words	430,379
Letter 1	0
Letter 2	5,223
Letter 3	31,441
Letter 4	91,201
Letter 5	128,811
Letter 6	96,270
Letter 7	49,755
Letter 8	19,671
Letter 9	6,118
Letter 10	1,603
Letter 11	398
Letter 12	79
Letter 13	16
Letter 14	2

4. PROPOSED USER INTERFACES

Since the IDE used for the prototype design of the suggestion system is Android, Android comes with XML file to create the user interfaces [25]. Thus, system interface, shown in Figure 4, has been prepared. Everything like font, text color, text style and other properties of a given text is set using xml file editor. The components which are user interfaces used in this prototype are Text View, Auto text completion. The systems modeling for a spellchecker and corrector system in mobile phone devices for the low-resourced Tigrigna language can be ready to be used. However, as it has been discussed, a user must write's the first word so as to inductee the suggestion system. The user interface shown in the Figure 4, is prepared to perform this operation.

Figure 4. Screen shot of the spellchecker and corrector system user interface

5. RESULTS AND DISCUSSION

The quality of design focused on ISO9241-10 usability metrics: ISO 9241 criteria's are introduced to guide the quality of design in this study. As a result of that, the results of how these criteria influence the design work are presented in this section. The spellchecker text entry has successfully created, and the suitability of the system modeling prototype based on the usability metrics. The result that achieved is found after the experiment shows in the Table 3, the number of relevant suggested and non-relevant (not correctly suggested) words for the corresponding word-lengths. The column named as collected words represents the number of collected words for testing the system with corresponding word length of each word. The column, Suggested Words shows the numbers of relevant words after word were written for word length. And the table shows the percentage of suggested words of the respective word-lengths.

Table 3. Number of relevant words and non-relevant words for each word length

Word Length	Collected	Word Length	Collected
2 Letter	100	100	100%
3 Letter	100	98	98%
4 Letter	100	97	97%
5 Letter	100	96	96%
6 Letter	100	64	64%
7 Letter	100	94	94%
8 Letter	100	93	93%
9 Letter	100	91	91%
Total	**800**	**733**	**92%**

In Table 3, 800 words of Tigrigna language are randomly collected as a sample to check the system modeling Tigrigna spellchecker accuracy from different word-length group. According to the researcher 733 (92%) of the total collected words were suggested by the application spellchecker and correction and the rest 67 (8%) are not suggested by the application. From this finding one come understand that accuracy of the systems modeling for Tigrigna spellchecker and correction in mobile phone devices scores 92%. This result shows clearly that the method is efficient in generating relevant suggested correct words and reduces the misspelled input words for writing Tigrigna words on mobile phone devices. Therefore this answers the statement of the problem this developed system model is efficient in generating relevant suggestion in spellchecker and correction. This question answers by generating relevant suggested words. Writing Tigrigna words to your mobile phone when we use the spelling checker and correction word entry method we can minimize Misspell input text and enhance the communication in writing Tigrigna language in our mobile phone devices.

6. CONCLUSION

A The objectives of this study are to design and develop model for Spellchecker and corrector for mobile phone devices for the low-resourced Tigrigna language, and subsequently test its actual performance based on ISO 9241 usability engineering standards. These were fulfilled whereby the Tigrigna spellchecker and corrector prototype has successfully corrected misspelled words during actual interaction by the Tigrigna users; thanks to techniques for correcting spell error: dictionary lookup, edit distance, and levenshtein distance.

The usability of this system modeling prototype based on the usability metrics (ISO 9241 usability engineering standards) achieving the result 92% is interpreted as Tigrigna users had effectively interacted with the system. Based on the foregoing the researchers, therefore, conclude that the techniques employed for correcting Tigrigna spelling error in mobile phone application text entry; and prototyping process model used to develop the system were all effective methods and procedures in building smart phone applications that involve text entry for Tigrigna people in Ethiopia. In the Future, The researchers are aware that the performance of the prototype system can be further enhanced by increasing the number of words in the database and evaluation carried out by the users of the language in mobile. We can also enhance the performance of the prototype systems by using hybrid approaches which combines edit distance with N gram techniques.

REFERENCES

[1] F. T. Bekele, "Morphology Based Spell Checker for Kafi Noonoo Language," Published Master's Thesis, Addis Ababa University of College of Natural Sciences, 2018.
[2] I. Scott MacKenzie and R. William Soukoreff, "Text Entry for Mobile Computing: Models and Methods, Theory and Practice," *Human-Computer Interaction*, vol. 17, no. 2, pp. 147-198, 2002.
[3] C. Whitelaw, B. Hutchinson, G. Chung, and G. Ellis, "Using the web for language independent spellchecking and auto correction," *EMNLP '09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, vol. 2, pp. 890-899, 2009.
[4] Q. Chen, M. Li, and M. Zhou, "Improving query spelling correction using web search results," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007, pp. 181-189.
[5] J. Gao, X. Li, D. Micol, C. Quirk, and X. Sun, "A large scale ranker based system for search query spelling correction," in *Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics*, 2010, pp. 358-366.
[6] P. Gupta, M. Sharma, K. Pitale, and K. Kumar, "Problems with automating translation of movie/TV show subtitles," *arxiv preprint AarXiv: abs/1909.05362*, 2019.
[7] H. Faili, E. Nava, M. Mortaza and P. M. Taher, "Vafa spell-checker for detecting spelling, grammatical, and real-word errors of Persian language," *Digital Scholarship in the Humanities Advance Access*, vol. 31, no. 1, pp. 1-31, 2014.
[8] M.R. Islam, M.R. Islam and T.A. Mazumder, "Mobile Application and its Global Impact," *International Journal of Engineering & Technology (IJET-IJENS)*, vol. 10, no. 06, pp. 72-78, 2010.
[9] Azath M., and Tsegay Kiros, "Statistical Machine Translator for English to Tigrigna Translation," *International Journal of Scientific & Technology Research*, vol. 9, no. 01, pp. 2095-2099, 2020.
[10] T. Semere, "Probabilistic Tigrigna-Amharic Cross Language Information Retrieval (CLIR)," Published Master's Thesis, Addis Ababa University of College of Natural Sciences, 2013.
[11] D. Thakur, "Prototyping Model in Software Engineering," computer notes 2019. [Online]. Available: https://ecomputernotes.com/software-engineering/explain-prototyping-model.
[12] Michael Deininger, Shanna R. Daly, Kathleen H. Sienko and Jennifer C. Lee, "Novice designers' use of prototypes in engineering design," *Design Studies*, vol. 51, pp. 25-65, 2017.
[13] S. Ellen, "Sciencing," 2018. [Online]. Available: https://sciencing.com/slovins-formula-sampling-techniques-5475547.html

[14] T. M. Miangah, "FarsiSpell: A spell-checking system for Persian using a large monolingual corpus," *Literary and Linguistic Computing*, vol. 29, no. 1, pp. 56-73, 2013.
[15] A. Mohammed, P. Pavel, S. Younes, S. Khaled and G. Josef, "Improved Spelling Error Detection and Correction for Arabic," *Proceedings of COLING 2012: Posters*, pp. 103-112, 2012.
[16] K. Shaalan, R. Aref and A. Fahmy, "An Approach for Analyzing and Correcting Spelling Errors for Non-native Arabic learners," *2010 The 7th International Conference on Informatics and Systems (INFOS)*, Cairo, 2010, pp. 1-7.
[17] V. Mottaiyan, "Bahasa Melayu predictive text entry for Short Message Service (SMS) on mobile phones," *Fakulti Sains Komputer dan Teknologi Maklumat*, Universiti Malaya, 2011.
[18] R. Mishra and N. Kaur, "A Survey of Spelling Error Detection and Correction Techniques," *International Journal of Computer Trends and Technology*, vol. 4, no. 3, pp. 372-374, 2013.
[19] Wilbur W. J., Kim W. and Xie N., "Spelling correction in the PubMed search engine," *Information Retrieval*, vol. 9, pp. 543-564, 2006.
[20] D. Blank, "Spelling Checking Algorithms: Design and Practice," in *Proceedings of the Spring Senior Conference, Bryn-Mawr College*, 2012.
[21] S. Rani and J. Singh, "Enhancing Levenshtein's Edit Distance Algorithm for Evaluating Document Similarity," *International Conference on Computing, Analytics and Networks*, vol. 805, pp. 72-80, 2018.
[22] J. Crowell, Q. Zeng, L. Ngo and E. M. Lacroix, "A Frequency-based Technique to Improve the Spelling Suggestion Rank in Medical Queries," *Journal of the American Medical Informatics Association*, vol. 11, no. 3, pp. 179-185, 2004.
[23] S. Luz and S. Sheehan, "Methods and visualization tools for the analysis of medical, political and scientific concepts in Genealogies of Knowledge," *Palgrave communications*, vol. 6, 49, 2020.
[24] J. Egbert and D. Biber, "Incorporating text dispersion into keyword analyses," *Corpora*, vol. 14, no. 1, pp. 77-104, 2019.
[25] Pichiliani, M. C., and Hirata, C. M., "Adaptation of Single-user Multi-touch Components to Support Synchronous Mobile Collaboration," *Mobile Networks and Applications*, vol. 19, no. 5, pp. 660-679, 2014.