DESIGN AND DEVELOPMENT OF MORPHOLOGICAL ANALYZER FOR TIGRIGNA VERBS USING HYBRID APPROACH

Hagos Gebremedhin Gebremeskel[1,2], Feng Chong[1,*] and Huang Heyan[1]

[1]Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications Laboratory, School of Computer Science and Technology, Beijing Institute of Technology, China
[2]Department of Computer Science, Mekelle University, Mekelle, Tigray, 231, Ethiopia.

ABSTRACT

Morphological analyzer is the base for various high-level NLP applications such as information retrieval, spell checking, grammar checking, machine translation, speech recognition, POS tagging and automatic sentence construction. This paper is carefully designed for design and analysis of morphological analyzer Tigrigna verbs using hybrid of memory learning and rules based approaches. The experiment have conducted using Python 3 where TiMBL algorithms IB2 and TRIBL2, and Finite State Transducer rules are used. The performance of the system has been evaluated using 10 fold cross validation technique. Testing was conducted using optimized parameter settings for regular verbs and linguistic rules of the Tigrigna language allomorph and phonology for the irregular verbs. The accuracy of the memory based approach with optimized parameters of TiMBL algorithm IB2 and TRIBL2 was 93.24% and 92.31%, respectively. Finally, the hybrid approach had an actual performance of 95.6% using linguistic rules for handling irregular and copula verbs.

KEYWORDS

Morphological Analyzer; Tigrigna verb Morphology; Machine learning; Character Based Analysis; Feature Extraction; Rules setting; Hybrid approach

1. INTRODUCTION

Language is one of the most fundamental aspects of human behavior which is an essential component of our daily lives [1]. It is also the mechanism by which information and knowledge can be kept for a long period of time and passed on from generation to generation. Language in its written form acts as a means of keeping recorded information and knowledge in the long term and transmitting what it records from generation to generation. In its spoken form, it works as a way to organize our daily life with others [2].

Linguistics is the study of languages, particularly natural languages. Natural language (NLP) is a sequence of conventions that humans use for communication[3]. Natural language processing is the academic discipline that studies the computer processing of natural language (NL). Morphology is a branch of linguistics that studies [4] the identification, analysis, and description of the structure of a language's forms and other linguistic units such as root words, suffixes, and parts of speech [5, 6]. Morphology attempts to formulate model rules for knowing the speakers of these languages.

DOI: 10.5121/ijnlc.2023.12501

NLP is a branch of computer science that studies the interactions between computers and human languages [7]. It is used to generate human-readable data from computer systems and to convert human language into more formal structures that a computer can understand [5]. As a result, it is significant for scientific, economic, social, and cultural reasons. Its theories and methods are disseminated in massive new language technologies, resulting in rapid growth. For this reason, a wide range of people must have a working knowledge of NLP. Within the industry, this includes people involved in human-computer interaction, business information analysis, and web software development. It includes academics from human computing and body linguistics to artificial intelligence and computer science [6]. The fundamental problems of NLP are morphological analysis, part of speech tagging, word sense disambiguation, and machine translation [5]. The morphological analyzer is the most important one of the NLP tools in the automatic processing of human languages. It analyses a text's naturally occurring word forms and identifies the root word and its features. Morphological analyzers are used in developing NLP applications such as machine translation, spell checkers, speech recognition, grammar checker, information retrieval, Part of speech tagging, automatic sentence construction, etc. Despite their importance, some languages, such as Tigrigna, need more publicly available morphological analyzers [8]. Therefore, this research paper focuses on developing Tigrigna verbs morphological analyzer.

Tigrigna belongs to the Semitic language family, is one of the four national languages of Ethiopia, Eritrea and the Tigray regional state of Ethiopia. This language is spoken by about seven million people worldwide and uses Ethiopic script for writing [9]. In addition to hard copy documents in libraries and documentation centers, there are enormous collections of Tigrigna documents on the web. Even as the number of documents increases, identifying the relevant documents related to a specific topic can take time and effort. The need for morphological analysis stems from the fact that natural languages are distinguished by morphological variations of words, which may take on multiple forms due to the addition of different affixes. The primary goal of morphological analyzer algorithms is to remove all the possible affixes, reducing the word to its stem [10] to make it more manageable.

The advent of personal computers and mobiles has increased communication between Tigrigna speakers using written texts. Many electronic documents are produced due to the worldwide communication. Those produced electronic documents need several automatic NLP systems as machine translation, information retrieval, grammar checking and the like to minimize the size of a word list to a manageable level and improve retrieval performance, and capture the strong relationships existing between different word forms in the language. For example, the word "በለዐ/ he eat" can exist in a document in several patterns that has same meaning such as "ምስበለዐ/after he eat", "በሊዑስ/ he eat then", "በሊዑከ/ he eat then" and other related words, but these are treated as different in the absence of document processing. For accurate document processing and retrieval, it is required to manage the root words. In such case, development of a morphological analyzer is required to show the similarity of these words by taking the root "በለዐ/ eat". This helps for the accuracy of information processing and extraction of Tigrigna documents. The accuracy of information aids to keep and transfer the identities to the next generation, citizens must know the meaning of these documents. In order to provide the necessary access to this wealth of information and enable its development, the basic information retrieval tools need to be designed and deployed. If users do not know the idea in the documents, they will not give any attention for heritages. These resources can also applicable as sources of philosophy, creativity, knowledge and civilization both to Ethiopia, Eritrea and all the world. To use these resources, one must be fluent in the language itself, or the literature must be manually translated into one of the currently spoken languages, which can be time consuming. Morphological analyzers for some international and Ethiopian languages as English, Dutch, Amharic, and Ge'ez

have been developed to address such issues, but no such system for Tigrigna verbs. This prompted us to design and develop a Tigrigna verb morphological analyzer.

Morphological analyzers have been developed for a different language, including English, Arabic, French, Spanish, Amharic, Ge'ez, Afaan Oromo, as well as others. An attempt has also been made for some Tigrigna feature verbs, adjectives and nouns morphological analyzer. Because the morphological properties of the Tigrigna language are generally complex, previous attempts to develop, a morphological analyzer for the language have been reported to have encountered numerous challenges. Furthermore, Tigrigna verbs' morphological properties are inherently more complex than those of other word classes.

Michael Gasser [11] attempted to developing morphological generation and analysis of Tigrigna feature verbs using finite state transducers with rule based approach, but not fully functional for derivative and inflectional verbs. Moreover, the system only works for few feature verbs. There is no effectively developed morphological analyzer algorithm that is implemented for those entire verb categories. Therefore, the aim of this study is to develop Tigrigna verbs morphological analyzer and generator of the entire verb categories.

2. LITERATURE REVIEW

Recently, morphological analyzers have gained a significant base in the advancement of natural language processing tasks. Design and development of morphological analyzer is a crucial tool for various NLP tasks as it enables the analysis and understanding of the language's morphological structure and patterns [12]. Morphology analysis tries to discover the rules that govern the formation of words from the smaller meaning bearing units, morphemes in a language [13]. Morpheme is the building block from which a word is made-up that could not be broken down further into meaningful parts [14]. The term morphology is normally attributed to the German poet, novelist, playwright and philosopher Johann Wolfgang von Goethe [15], who coined in the early nineteenth century in a biological context. The word comes from the Greek “morph,” which means “form, shape,” and morph is the study of shape or shapes. Morphology in linguistics refers to the mental system involved in forming words or to the branch of linguistics deals with words, their internal structure and how they are formed [15]. A primary source of information about morphology is formed by the descriptive grammars of individual languages which usually give a description of inflection and word formation. As mentioned in Chapter 1, morphology is the branch of linguistics concerned with the internal structure of words [16]. It is the study of word formation - how words are made up of smaller parts. When we do morphological analysis, we ask questions like what bits does this word contain? What do all of them mean? How are they combined?

Morphology is the study of the internal structure of the word. Morphological analysis retrieves the grammatical features and characteristics of a morphologically associated word [17]. The morphological analyzer is a computer program that takes a word as an input and produces its syntactic structure as an output. It will return its root / origin along with its grammatical information depending on the word class [18].

Tesfaye Bayu [19] designed and developed a morphological analysis system for Amharic language using a Linguistica. Linguistica is a freely available software package which is designed for analysis of morphology. As the author indicated the package requires a large corpus ranging from 5,000 to 1,000,000 words. However, the author used a 5,236 words corpus, the smallest recommended corpus size, to train the morphology of Amharic using Linguistica. Lingustica2001 [19] is designed for concatenative morphology of languages but it is not appropriate for non-concatenative morphology of languages like Amharic. Therefore, the author developed a stem

internal morphological parser (called Amharic Stems Morphological Analyzer ASMA) based on the theory of auto-segmental morphology theory to analyze the stems identified by Linguistica into their constituent root and pattern morphemes. Unfortunately, the author could not integrate the stem analyzer with Linguistica due to time limitations. The author suggests conducting further research using a different approach to develop an efficient morphological analyzer.

Wondwossen Mulugeta and Gasser [20] used Inductive Logic Programming to develop a supervised machine learning approach to morphological analysis of Amharic verbs (ILP). This approach draws hypotheses from background knowledge and examples and represents them in the form of logic programming. New instances are assessed by using the hypothesis and background knowledge as a basis for evaluation. The authors tested it using CLOG which is a Prolog based ILP system. The system primarily learns first-order decision lists or rules based solely on positive examples. These rules in the CLOG consist of left and right clauses, where the right side represents a condition and the left side represents the conclusion. For a rule to be considered true, all conditions must hold true. The researchers manually labeled 206 simple verbs for training and focused on training Amharic verb stem extraction, internal alternation, and roots separately within the CLOG. Their training efforts resulted in the extraction of 19 root templates and 108 stem templates, and when combined, they achieved an accuracy of 86.99% in their testing. They created a training set comprising 1,784 Amharic verbs, but their investigation was limited to simple Amharic verbs, specifically subject markers in both prefixes and suffixes. The researchers mentioned that ILP (Inductive Logic Programming) is suitable for morphological analysis in languages like English that exhibit simpler morphological structures. However, with the inclusion of sophisticated background predicates and a larger number of examples, ILP could also be applicable to complex languages. They suggested the possibility of conducting studies on complex Amharic verbs and other word categories within the language. It is important to note that CLOG is incapable of learning rules from incomplete examples. Therefore, ILP with CLOG requires complete examples in order to learn morphology rules and analyze new instances of a given word. However, in reality, this is not practical, as it would be impossible to expect every possible combination of thousands of morphemes to appear in the training set, particularly for agglutinative languages like Amharic and Ge'ez [20].

Michael Gasser [21] developed a morphological analyzer and generator for the three Ethiopian languages namely Amharic, Oromo and Tigrigna. The Analyzer and generator focus on verbs of the three languages and including nouns for Amharic. The analyzer segments words into their component morphemes and assign grammatical morphemes to grammatical categories and lexical morphemes to lexemes. For example, given an Amharic word, HornMorpho, returns the root, the lemma and a grammatical analysis in the form of a feature structure description for each possible analysis. On the other hand, the morphological generation performs the reverse process. The author derived lexicons for the three languages from online dictionaries. For Amharic, as the author stated the lexicon is derived from the Amharic English dictionary of Aklilu which contains 1,851 verb roots and 6,471 noun stems. For Oromo, the lexicon of verb and noun roots are extracted from the dictionaries of Gragg and Bitima that contains 4,112 verb roots and 10,659 nouns stems. Likewise, for Tigrinya, the lexicon of verb roots is derived from “Efrem Zacarias“ around 602 verb roots. The system was implemented using finite state transducer and evaluated with 200 Amharic and 200 Tigrigna verbs, and 200 Amharic nous and adjectives. Each word was selected randomly. The system was run on those words and the results were evaluated by a human reader who is familiar with the languages made 8 (96% accuracy) and 2 (99% accuracy) errors for Tigrigna and Amharic verbs respectively. For Amharic nouns and adjectives, it made 9 errors (95.5% accuracy).

Kibur Lisanu [22] studied morphological synthesis of Amharic perfective verb forms. A prototype was developed using rule based and artificial neural network approaches. The rule

based approach generates all the roots successfully whereas the neural network predicts the type of roots in the test dataset with an accuracy of 81.48%. The verbs were classified in to three categories as type A, B and C. Each type tested separately using the neural networks and achieved an accuracy of 80%, 25% and 100% respectively. As stated in the paper the developed a system named Amharic Morphological Synthesizer (AMS). The author only considered Amharic verbs particularly perfective forms.

Mesfin Abate and Yaregal Assabie [23] developed a morphological analyzer for Amharic language using MBL. The corpus contains 1022 words of which 181 and 841 are nouns and verbs respectively. As the author stated the number of instances extracted from nouns and adjectives are 1356 and verbs are 6719 which accounts a total of 8075 instances. Within these instances, 26 different class labels occur. The experiment was conducted on the handcrafted dataset on TiMBL 6.4.4 with default settings and by tuning the different parameters. Based on the default values of parameters of each algorithm (IB1 and IGTREE), experimental results show that the generalization performance of IB1 and IGTREE algorithms are 92.02% and 76.27% respectively. Similarly, with optimized parameters 93.59% and 82.26% results were obtained for IBl and IGTREE respectively. The author also used the default parameter settings for IB1 algorithm with leave one out and 10fold cross validation and the generalization accuracy of the model obtained were 93.3% and 92.02% respectively. This shows that generalization performance of the learned model is almost the same by the two evaluation methods. The 10fold evaluation results of IB1 and IGTREE on optimized parameter settings are 93.59% and 82.26% respectively. However, the performance of IB1 on optimized parameter settings is raised to 96.4% when evaluated with LOOCV. This happens since LOO evaluation uses all the dataset for training except one, which helps the model to learn better. From this, we can conclude that IBl has a better performance than IGTREE even if it consumes more memory and time than IGTREE.

Desta Berihu [24] carried out research and development work on morphology of Ge'ez verbs using rule based approaches specifically CV based and Two Level Morphology (TLM) to design the model and to implement the prototype of the analyzer. However, Desta limited to only ቀተለ /qätälä/ category verb forms among the eight main verbs. The system performs an accuracy of 92.05% at feature level and of 73.98% at verb level.

Hagos Gebremedhin and Frank Chao Wang [25] developed a morphological analyzer that covers all root category verb forms: ቀተለ /qätälä/; ቀደሰ፣ ደንገፀ፣ ባረከ፣ ማህረከ፣ ሴሰየ፣ ክህለ፣ ጦመረ and multi radical root patterns using machine learning approach and the overall accuracy with optimized parameters using IB2 and TRIBL2 was 93.24% and 92.31% respectively. Similarly, the overall precision, recall and F score with optimized parameters using IB2 were 55.6%, 56.3% and 59.95% respectively. In the same manner the precision, recall and F score using TRIBL2 were 58.8%, 60.3% and 59.54% respectively.

3. DESIGN AND DEVELOPMENT METHODS

3.1. Architecture Design

Morphological analysis of highly inflected languages like Tigrigna are nontrivial task. The design and development used in this paper is hybrid of memory-based learning and rule based approach as seen in Figure **1** below the general system architecture. The memory based learning approach of morphological analysis primarily concerns saving or learning of some patterns of the morpheme in memory and trying to classifying and analyzing the newly or unseen words by analogy on the rule based approach then transfer to the analysis phase. Finally, it displays morpheme with function as final output.

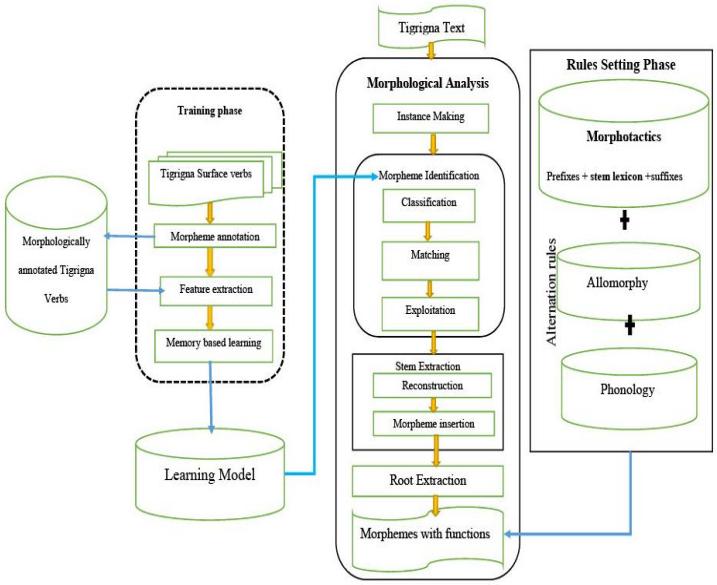


Figure 1: General Architecture of Tigrigna verbs morphological analyzer

3.2. Training Phases

The training phase of this study is based on the training of memory based learning process have the morpheme annotation, feature extraction and memory based learning processes, and two knowledge base data storages through accepting the surface verb as input.

3.2.1. Morpheme Annotation

Morpheme annotation is the preparation of verbs by adding notes for explanation or comments for analysis the language verb pattern behavior. Tigrigna have five verb families with six stem patterns of CVCVCV, CVCCV, CVCVV, CVVCV, CVCCVCV, and CVCCVV on annotating the morphemes. Depending on these families and stem categories, we have annotated them manually. In this annotation process the system uses the following tasks for identifying and performing in the order listed.

- Accept surface verb
- Identifying inflected words
- Segmenting the word in to prefix, stem, suffix and circumfix
- Putting boundary marker between each segmented words
- Describing the representation of each markers
- Deposit the annotated Tigrigna Verbs

Tigrigna verbs have four Segments for prefixes before the stem and two segments for suffixes after the stem.

NegPref/PosPre/SMS/OMS + **Stem** + SufCirc/NegSuf

3.2.2. Feature Extraction

After annotated the verbs are stored in a database, features are extracted automatically from the manually created morphological database to make instances using Algorithm 3.1 based on the concept of windowing method in a fixed length of left and right context which is the average word length in the database. Windowing method is dividing the windows where the instances are placed in the left and right context to hold fixed length string of features, which describe the linguistic context of the token to be classified. Each instance is associated with a class. The class represents the morphological category in which the given word possesses. The feature extraction using windowing method in this paper can be calculated as seen in Equation (1) below:

$$\Delta(X, Y) = \sum_{i=1}^{n} \delta(xi, yi) \quad (1)$$

$$\text{Where: } \delta(xi, yi) = \begin{cases} abs\left(\frac{xi - yi}{maxi - mini}\right) & \text{if numeric, else} \\ 0 & \text{if xi} = yi \\ 1 & \text{if xi} \neq yi \end{cases}$$

The dataset for the learner consists of description of instances in terms of fixed number of feature values. Each character is used once as a focus character (F) and associated with the ten characters to its left (L1→L10) and the ten characters to its right (R1→R10). Each feature is separated by comma to be used as input for the learner. TiMBL supports different formats for the training and test files, including Compact, C4.5, ARFF, Columns, Sparse and Binary. In this study, the C4.5 format is the default method by which the training data is presented to TiMBL. C4.5 format implies that feature values are separated by commas and that the last feature value denotes the class of the instance. Therefore, it requires feature values and classes as shown in Algorithm 1 below.

Input: Inflected words
Output: -Instances in a fixed-length of vector size.

1. Define the window size length.
2. Fix the middle positions of arrays as a focus letter (the focus character represents where a character is started from that position on words).
3. Read from the KB and push one step forward each character until the right context Reached.
4. Put zero at the class if there is no any number and capital letters, next to the characters placed in the focus letter; if any one of those symbols exist put the value as a class (in last index)
5. Push the previous focus letter to the left and start putting each letter (as in step 3)
6. Go until it finishes that line
7. Go to the next line and repeat 3,4,5,6.

Algorithm 2: Feature Extraction

3.2.3. Memory Based Learning

Memory Based Learning learn based on the hypothesis that behavior can be extrapolated from stored representations of previous experiences to new situations based on the similarity of the old and new situations. MBL algorithms take a set of examples (fixed length patterns of feature

values and their associated class) as input, and produce a classifier that can classify new, previously unseen, input patterns.

3.3. Rule Setting Phase

This phase is for controlling the analysis of some verbs with the expert rules, which access the lexeme and use the morphotactics that could not handle in corpus.

3.4. Morphological Analysis Phase

The training and rule setting phases are the base to implement the morphological analysis phase. This phase includes instance making to make the input words to be suitable for memory based learning classification, the morpheme identification to classify and extrapolate the class of new instances, the stem extraction to reconstruct and insert identified morphemes, and finally the root extraction to get root forms of verb stem with their grammatical functions.

4. EXPERIMENT AND RESULTS

The experiment conducted to evaluate the performance of the analyzer is presented below on both the memory based learning and hybrid approaches. The verbs used in this implementation include all the inflected, derivated and compound words for all the regular verbs and irregularities including the copula verbs.

4.1. Machine Learning Experimental Result

The experimental result of the memory based learning was tested on the IB2 and TRIBL2 algorithms via default and optimized parameter settings. As shown in Table 1 the performance of IB2 algorithm with default and optimized parameter setting is 91.72% and 93.24% respectively. Similarly, the performance of TRIBL2 algorithm with default setting is 91.19% and with optimized parameters is 92.31%.

Table1: Performance of 10 fold Experiment with Optimized Parameter Setting

Evaluation method	Algorithm	Compression (%)	Time taken (seconds)	Size of instances base (byte)	Accuracy (%)
10 fold CV	IB2	56.66	0.263	7944	93.24
10 fold CV	TRIBL2	52.4	0.082	8112	92.31

The IB2 algorithm shows better performance than TRIBL2 in both default and optimized parameter settings. It also performs better in compression the instance trees and uses less memory than TRIBL2. On the other hand, TRIBL2 processes within short seconds than IB2 on the same number of instances.

4.2. Hybrid Approach Experimental Result

The hybrid of memory based learning and rule based approaches were tested. The analyzer was tested with 598 word stems and a corresponding morphologically derivated and inflected verbs, 143 irregular and 25 copula verbs. The system was tested with a total of 8112 verbs. This included, the derivational, inflectional, compound and copula verbs. Performance of the analyzer was measured based on the number of correctly analyzed verbs (as shown in Equation 2 below).

Basically, it was calculated as the number of correctly analyzed verbs divided by the total number of testing verbs multiplied by 100 to be expressed in percentile.

$$\text{Accuracy} = \frac{\text{Total Number of Correctly Analyzed Verbs}}{\text{Total Number of TestingVerbs}} \quad (2)$$

Generally, the analyzer accepts a morphologically complex (inflected) surface strings and derivated verbs of all verb categories, and it produces grammatical lexical strings. To measure performance of this paper, the formula of performance or accuracy stated below was used.

Table2: The performance result of the Hybrid approach morphological analyzer

Word type	No of input words	Correctly analyzed	Wrong analyzed	Performance /Percentage/
Regular Verbs	7944	7596	348	95.6%
Irregular Verbs	143	131	12	91.5%
Copula	25	24	1	96%
Total	8112	7751	361	95.6%

The new proposed model was experimented, morphological analysis with 10 fold cross validation. The system is trained on approximately 90% of the corpus and then tested on the remaining 10%. The performance of the system in terms of accuracy, time classification and memory usage was determined by evaluating the morpheme identification through training the system with the default and optimized algorithmic parameter settings. The evaluation criteria used for morpheme identification were accuracy, recall, precision, and F-score.

The precision, recall and F score were also calculated by taking the average of the 10 fold cross validation. The results using IB2 algorithm with default parameter settings are 52.9%, 52.1% and 52.49 %, respectively. Similarly, TRIBL2 classifier was also evaluated in the same manner with IB2 and obtained 55.4%, 56.6%, 55.99% precision, recall and F score, respectively. These algorithms are also evaluated using optimized parameters to obtain good result than the default ones. Therefore, we obtained 55.6%, 56.3% and 59.95% precision, recall and F-score respectively using IB2 algorithm. In the same manner we obtained 58.8%, 60.3% and 59.54% precision, recall and F score respectively using TRIBL2 algorithm. In general, both algorithms have insignificant difference. Therefore, they are applicable for Tigrigna morphological analysis hybridlly.

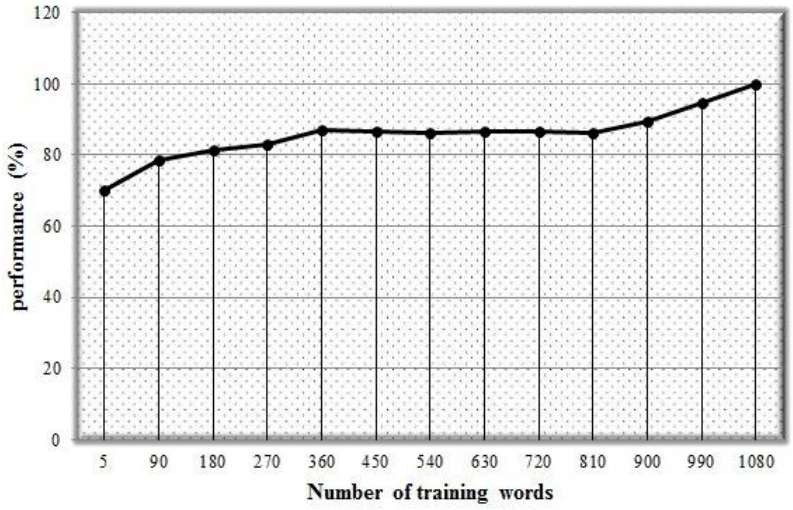


Figure 2: Learning Curve with Increasing Number of Words

5. CONTRIBUTIONS OF THE PAPER

Morphological analyzer is the base for every high level applications. The main contributions of this paper work are listed as follows:

- New corpus and have been developed for Simplifies the barrier of the users of the language then fills communication gap among end users.
- The study has adopted approaches of morphological analysis system of other languages for Tigrigna language and designed approaches, algorithms, and techniques in developing automatic morphological analysis of Tigrigna verbs.
- The study identified and proposed a hybrid of memory based learning and rule based system architecture with linguistic rules of the language to handle irregular and copula verbs allomorphy and phonology.
- The study has designed automatic morphological analyzer system prototype that analysis the morphology of Tigrigna verbs.
- The study identified basic challenges in developing and implementing automatic Tigrigna verb morphological analysis system and propose the possible approaches to solve the complexities.

6. CONCLUSION

In this paper, the hybrid of memory based learning and rule based analyzer for Tigrigna verbs are used. Based on the experimental results obtained in the previous chapter, the hybrid of memory based learning and rule based approaches showed a good result for morphological analysis of Tigrigna verbs relative to small number of datasets. The unavailability of complete inflection verbs of all Tigrigna verb categories and annotated morphological database forced us to spend much more time in preparing dataset. This is also the main reason for the small number of our datasets. We annotated manually 589 stem verbs to be suitable to TiMBL algorithms. From these annotated verbs, we extracted 8112 instances automatically. This data set was divided into training and testing data from which 90% for training and 10% for testing. The training data is used to assess how much the model is able to learn and the test data used for evaluating the performance of the algorithms. By default and adjusting optimized parameter settings of TiMBL tools, we trained and tested our dataset. To do this, IB2 and TRIBL2 algorithms are used. We

found that IB2 is good at memory usage on both default and optimized settings (with 91.72% and 93.24% accuracy) in the learning model but it has low processing speed which in turn takes more time. On the other hand, TRIBL2 learning model algorithm performs a little bit different from IB2. It performs 91.19% and 92.31% with default and optimized parameter settings respectively. TRIBL2 classifier needs more memory usage and high speed during training and test of dataset. Since there is tradeoff between both algorithms is respective of their advantages. Thus, memory storage and speed may have a matter in choosing from both algorithms for Tigrigna morphological analysis. In the system, some verbs are handled using the linguistic rules allomorphy and phonology of the language by a lexicon developed in finite state transducer to make the system more efficient analysis. The hybrid analyzer have a general performance 95.6% through integrating the rule based analysis of irregular and copula verbs. Therefore, the hybrid approach is efficient for Tigrigna verbs morphological analyzer and have better performance.

The study recommend the as future works a comparative analysis for Tigrigna and other local languages, using other approaches such as HMM, SVM, MBL, ILP and finite state morphological analysis can improve and to make a complete full flagged morphological analysis.

ACKNOWLEDGMENTS

This research is supported by Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications laboratory of Beijing institute of Technology. Special thanks to my Supervisors professor Huang Heyan and Feng Chong.

REFERENCES

[1] A. S. Morcos, D. G. Barrett, N. C. Rabinowitz, and M. Botvinick, "On the importance of single directions for generalization," *arXiv preprint arXiv:1803.06959*, 2018.

[2] J. Allen, *Natural language understanding*: Benjamin-Cummings Publishing Co., Inc., 1995.

[3] A. Adrian, A. Richard, F. Ann, and M. Robert, *An Introduction to Language and Communication*, 4th ed. New York: University Press, 2003.

[4] B. Srinivasa-Desikan, *Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python, Gensim, spaCy, and Keras*: Packt Publishing Ltd, 2018.

[5] K. Ak and O. T. Yildiz, "Unsupervised morphological analysis using tries," in *Computer and Information Sciences II*, ed: Springer, 2011, pp. 69-75.

[6] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*: " O'Reilly Media, Inc.", 2009.

[7] J. J. Bird, A. Ekárt, and D. R. Faria, "Chatbot Interaction with Artificial Intelligence: human data augmentation with T5 and language transformer ensemble for text classification," *Journal of Ambient Intelligence and Humanized Computing*, 2021/08/23 2021.

[8] S. Lushanthan, A. Weerasinghe, and D. Herath, "Morphological analyzer and generator for Tamil language," in *2014 14th International Conference on Advances in ICT for Emerging Regions (ICTer)*, 2014, pp. 190-196.

[9] J. Hammond, A, "Chronicle of the Revolution in Tigray region of Ethiopia," Red Sea Press., Eritrea1999.

[10] J. Dawson, "Suffix removal and word conflation," *ALLC bulletin*, pp. 33-46, 1974.

[11] M. G. R, "Optimal inflection in Tigriya: A constraint-based to a non concatinative," in *univesity of Toronto masters of Art Department of linguistics*, 2011.

[12] D. Harvey, F. Lobban, P. Rayson, A. Warner, and S. Jones, "Natural language processing methods and bipolar disorder: scoping review," *JMIR mental health*, vol. 9, p. e35928, 2022.

[13] Koskenniemi, "a general morphological processor". In Dalrymple, M., Doron, E., Goggin, J., Goodman, B., and McCarthy, J., editors, Texas Linguistic Forum " *Department of Linguistics, The University of Texas at Austin, Austin, TX.*, vol. 22, pp. 165-186.

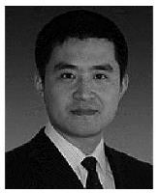
[14] Martha-Yifiru. (2010, August) Morphology-Based Language Modeling for Amharic. *Addis Ababa Äthiopia*.

[15] M.-A. a. Kirsten-Fudeman, *What is Morphology?,2nd Edition*: Blackwell Publishing, 2011.
[16] Geert-Booij. (2009) The Oxford Handbook of Grammatical Analysis Oxford *Oxford University Press*. 563-589.
[17] D. s. Anand Kumar, and Krishnan Soman, "A Sequence Labeling Approach to Morphological Analyzer for Tamil Language," in *Tamil University Department of Linguistics*, Thanjavur, India., 2010.
[18] Saranya-Kittanakom, "Morphological Analyzer for Malayalam Verbs, a project report submittedin partial fulfillment for the award of the degree of Master of Technology in Computational Engineering and Networking, Amrita Vishwa Vidyapeetham Amrita School of Engineering," Coimbatore, 6411052008.
[19] Tesfaye-Bayu. (2002) Automatic morphological analyzer for Amharic: An experiment employing unsupervised learning and auto segmental analysis approaches. *Master''s thesis Addis Ababa University*.
[20] W.-M. a. Michael-Gasser, "Learning Morphological Rules for Amharic Verbs Using Inductive Logic Programming," in *Workshop on Language Technology for Normalization of Less-Resourced Languages SALTMIL8/AfLaT2012)*, 2012.
[21] M. Gasser, "HornMorpho: a system for morphological processing of Amharic Oromo, and Tigrinya," in *Conference on Human Language Technology for Development*, Alexandria, Egypt., 2011.
[22] Kibur-Lisanu. (2002) Design and development of automatic morphological Synthesizer for Amharic perfective verb forms. *A thesis submitted in partial fulfillment of the requirement for the degree of masters of science in information science, Addis Ababa University*.
[23] Mesfin-Abate, "Amharic Morphological Analysis Using Memory Based Learning," *Thesis Submitted in Partial Fulfillments of The Requirement for the Degree of Master of Science in Computer Science, Addis Ababa University, Ethiopia*, 2014.
[24] D. Berihu. (2010) Unpublished Design and Implementation of Morphological Analyzer for Ge'ez Verbs. *Master's Thesis, Department of Computer Science: Addis Ababa University*.
[25] G. H. Gebremedhin and F. C. Wang, "Morphological Analyzer for Ge'ez Verbs Using Memory Based Algorithms," *International Journal of Technologies* pp. 1-13, Julay 30 2020.

AUTHORS

Hagos Gebremedhin Gebremeskel, Ph.D. candidate, received master's degree in Software Engineering from the School of Software, Nankai University, Tianjin, China, in 2021. He is currently pursuing the Ph.D. degree at the School of Computer Science and technology, Beijing Institute of Technology, Beijing, China. His research interests include natural language processing and deep learning, particularly Tigrigna Large Language models.

Feng Chong (M'17) received the Ph.D. degree in computer science from the University of Science and Technology of China, Beijing, China. He was an Associate Researcher with Chinese Academy of Sciences, Beijing, China from 2005 to 2010. Associate professor from 2010 and he is currently a professor at Beijing Institute of Technology, Beijing, China. His research interests include natural language processing, in particular machine translation, computer aided translation, social media processing and knowledge graph.



Heyan Huang received her Ph.D. degree in institute of computing technology from the Chinese Academy of Sciences, Beijing, China. She was a Researcher and Director with the Language Information Engineering Research Center, Chinese Academy of Sciences, Beijing, China from 1997 to 2008. From 2009, she has been the Dean and a Professor of the School of Computer Science at Beijing Institute of Technology. Her current research interests include natural language processing, machine translation and artificial intelligence.