MODELING AND IMPLEMENTING AMHARIC NON-STANDARD WORDS SPELLING CHECKER AND CORRECTOR

A Thesis Presented

By

MESERET MOSSIE

To

The Faculty of Informatics

Of

St. Mary's University

In Partial Fulfillment of the Requirements for the Degree of Master of Science

In

Computer Science

January, 2022

ACCEPTANCE

MODELING AND IMPLEMENTING AMHARIC NON-STANDARD WORDS SPELLING CHECKER AND CORRECTOR

By

MESERET MOSSIE

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

_____

Internal Examiner

_____

External Examiner

_____

Dean, Faculty of Informatics

January, 2022

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been duly acknowledged.

_____

MESERET MOSSIE

_____

Signature

Addis Ababa

Ethiopia

This thesis has been submitted for examination with my approval as advisor.

_____

HAFTE ABERA

_____

Signature

Addis Ababa

Ethiopia

January 15, 2022

ACKNOWLEGDMENTS

First and foremost, I would like to thank the Almighty God for giving me the health, wisdom, strength and opportunity to undertake this research study. Without his blessings, this achievement would not have been possible.

I would also like to thank my advisor, **Mr. Hafte Abera** for his encouragement and constructive comments starting from the beginning to the end of this study.

Besides, I would like to express my deepest gratitude to **Mr. Haben B. (PhD candidate)** who helped me in structuring the organization of my thesis relentlessly. Similar gratitude goes to those who filled my questionnaire important for evaluation of my work responsibly.

My thanks are also extended to all my family for their companionship in the course of doing this thesis. Similar thanks are also due to my parents, brothers and sisters, and friends for their moral encouragement.

TABLE OF CONTENTS

LIST OF ACRONYMS

ASR	Automatic Summarization, Machine Translation, Speech Recognition
IDE	Integrated Development Environment
GUI	Graphical User Interface
IR	Information Retrieval
ISO	INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
OCR	Optical Character Recognition
SQL	Structured Query Language
SPYDER	Scientific Python Development Environment
TCL	Tool Command Language
TK,	Toolkit

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Amharic is a language that is spoken by millions of people in Ethiopia and Ethiopian living internationally. It is a widely used language for creating documents for communication purposes. However, since no spelling checker computer program detects and corrects for non-standard Amharic language, spelling errors are becoming common and interfere with communication. A spelling checker is a computer program that detects and often corrects misspelled words in a text document. In response to this problem, the researcher sets developing and implementing nonstandard error Amharic language spell checker and corrector model. To achieve this objective the researcher uses design science research methodology. This research is aimed at modeling and designing Amharic non-standard words spell checker and corrector, and hence research of this nature is best addressed through design science research methodology.

The researcher uses dictionary lookup for error detection technique and minimum edit distance as error correction technique. While dictionary lookup detects misspelled words sequence matcher provides spelling suggestions and the list of candidate spellings. The research also employs tools for compiling the python code and storing the corpus. It also uses tools for text processing and for developing a graphical user interface.

To demonstrate the validity of the non-standard Amharic words spelling checker and corrector model and to measure its accuracy, precision and recall, confusion matrix have been used as measuring matrix. As a result the model precision, accuracy and recall has 0.94, 0.93 and 0.87 respectively. A questionnaire is also prepared to measure the usability of the prototype on basis of ISO 9241 usability engineering standards and distributed to respondents who are familiar with Amharic Writings. Based on this, the research finds out that the accuracy of the model designed for non-standard words Amharic language is 92%. This clearly shows that the model is effective in checking and correcting words written in Amharic.

Key words:

Amharic non-standard words, spell checker, correction, suggestion, Edit Distance, Dictionary Lookup

CHAPTER ONE

INTRODUCTION

1.1 Back Ground of the Study

Amharic is Ethiopia's national language, descended from Geez, and has been spoken in Ethiopia since the 4th century AD. It is on its way to become the world's second most extensive semantic language, behind Arabic. In general, Ethiopia has more than 86 languages spoken by a population of more than 90 million people. In the Ethiopian context, Amharic scripts were the first to be utilized for industrial, economic, and political purposes; as a result, they indicated in the preceding paragraph that Amharic is Ethiopia's official language [1].

Natural language processing (NLP) is a theory-motivated range of computational techniques for the automatic analysis and representation of human language. NLP research has evolved from the era of punch cards and batch processing (in which the analysis of a sentence could take up to 7 minutes) to the era of Google and the likes of it (in which millions of webpages can be processed in less than a second) [2].

Spell checking is the process of finding misspelled words in a written text, and possibly corrects them. It can be either stand-alone application capable of processing a string of words or a text or as an embedded tool which is part of a larger application such as a word processor [3]. Various search and replace algorithms are adopted to fit into the domain of spell checker. It identifies the words that are valid in the language as well as misspelled words in the language and suggests one or more alternative words as the correct spelling when a misspelled word is identified.

Spelling checker provides two core functionalities; spelling error detection and spelling error correction. “Error Detection” is verifying the validity of a word in the language while “Error Correction” is suggesting corrections for the misspelled word [4].

Spell checker can also be of two types; interactive and automatic. In the interactive spellchecker can suggest more than one correction for each error and the user has to select one for replacement on the other hand in automatic correction, the spellchecker has to decide on the one best correction and the error is automatically replaced with it. This is chosen for those speech processing and Natural Language Processing (NLP) related systems where human intervention is not possible [4].

The spell checking process has three sequential components. Detecting errors is the first component and finding correction and ranking correction are tlhe second and third components respectively. While detection and correction are already defined in the above paragraph ranking is the ordering of suggested corrections in decreasing order of their likelihood for being actual intended word. Spelling error correction can be grouped in to Non-word error detection, Isolated word error correction and Context dependent error correction [5]. Identifying and correcting non-standard words categorized as isolated word error detection. Spell checking is not new in the areas of Information Retrieval and Language processing.

Many different techniques for detection and correction of spelling errors are proposed since 1960s. Some of these techniques exploit general spelling errors trends while others use the phonetics of misspelled word to find likely correct words [5].

Quite a few of these techniques are being used with text editors and other text handling applications and are showing reasonably good performance. The problem of spell checking is still considered open for further research and improvements. This is due to the fact that the research in the area of Natural Language Processing advanced over the years; the need of spell checking is being felt for many tasks other simple proof reading of computer-generated text [6]. Besides, spell checkers are widely used in other applications such as Optical Character Recognition systems, Automatic Speech Recognition systems, Computer Aided Language Learning Software, Machine Translation systems and Text-to Speech systems [7]. The second reason for considering the spell-checking problem unsolved is that most of the techniques proposed so far are based on English or some other Latin script-based language. Since every language has its own writing system, the techniques that perform well for one language may not perform that well for some other language; they may even totally fail [4].

These days, using Amharic language for writing documents is growing in a very fast velocity. Government officials and departments, legal institutions, business offices, media channels, schools all use desktop and personal computers in their daily work. As writing documents through this language is advancing fast, spelling errors which interferes communication between the writer and reader are also mounting. This is due to the fact of being human in that doing mistakes while creating documents is often unavoidable.

Though the use of this language is growing fast, this language lacks even one very basic non

standard spell checker and corrector. Hence, building non-standard Amharic word spell checker and corrector have an outstanding effect on Amharic language processing applications. Therefore, in response to this, this research is aimed at developing Non-Standard Amharic word spelling checker and corrector through minimum edit distance technique to alleviate the problem.

1.2 Statement of the problem

Language is an important tool to facilitate communication between persons, business organizations and government institutions. In this contemporary world, written communication through different languages is widely used. However, since spelling mistakes are inevitable while creating documents for communication purposes, researchers have modeled and developed spelling checker and corrector application for different languages. Spell checker is a computer program that detects and often corrects misspelled words in a text document [8]. It provides error detection and correction functions [9]. It can be a standalone application or an add-on module integrated into an existing program such as a word processor or search engine [4].

The use of Amharic language in creating documents is growing fast. It is widely used language in the whole Ethiopia and some city of the world [10]. In Ethiopia, it is the national official working language. Different government institutions, such as federal offices, schools, justice offices, courts, Medias and others, private businesses such as law offices, businesses, and private media institutions use Amharic language for creating documents through desktop and personal computers.

Despite the growing use of this language for creating documents through desktop and personal computers, spelling errors which interferes communication between the writer and reader are also mounting. This is because spelling mistakes while creating documents is natural associated to the fact of being human and due to the absence of spell checker and corrector for non-standard Amharic words.

Writing documents through Amharic language are confronted with time consuming task on editing errors especially with the non-standard words. Though huge time is spent on editing

errors there is no guarantee that all errors are fixed. Blind and low vision users may also face similar problems. This ultimately leads to miscommunications between the writer and reader.

Most of the local dictionary did not include non-standard Amharic words. But people use non-standard words in their written materials. At this time if the spellchecker is unable to detect and correct those words it would be a huge effort to detect manually. The researches[11], [12][15] uses only real word and Non-word spell checker in their work.

The researcher has exerted maximum effort to search whether similar works have been done in this area to avoid redundancy. However, no work on non-standard Amharic words error spelling checker and corrector is found. The spelling errors occurred during creating documents through Amharic language together with the absence of non-standard words spell checker and corrector for Amharic language motivates the researcher to search on this area. Therefore, in response to the above problems, this research is aimed at developing non-standard word Amharic language spell checker and corrector released as a stand-alone application that can be used by everyone who uses desktop application to identify and correct misspelled words in documents written in Amharic.

1.3 Research questions

- What is suitable approaches for non-standard word spell checker and corrector?
- What model should be used for non-standard word Amharic language spell checker and corrector?
- How to develop the prototype for Non-standard word Amharic language spell checker and corrector?

1.4 Objectives of the study

1.4.1 General objective

The general objective of the study is to design and implement a model for non-standard Amharic words spelling checker and corrector.

1.4.2 Specific objectives

- To review literatures and related works
- To study the nature of the language
- Preparing non-standard Amharic word corpus and evaluation and/or testing.

- To study approaches for non-standard word spell checker and corrector
- To propose a model for non-standard Amharic word spell checker and corrector.
- To develop a prototype for non-standard Amharic word spell checker and corrector.

1.5 Significance of the study

This research which is intended to develop non standard Amharic word spelling checker and corrector has the following significances;

- It will benefit for those who use desktop computer in writing documents through Amharic language to shorten time for editing.
- It can also use as an input of reference for potential researchers who have an interest to conduct research on this area.
- This also benefits to the field of computer science for this research adds something to the existing body of knowledge in computer science.

1.6 Scope of the study

The study is aimed at developing Non-standard Amharic words Spelling Checker and Corrector through dictionary based minimum edit distance (sequence matcher algorithm) technique by employing tools like python programming language and Notepad to store words. As it has been said in the background part, spell errors are categorized into Non-word errors, non-standard and real word errors, and hence the scope of this research is confined to Non-standard Amharic word errors.

1.7 Limitation of the study

Conducting research is a big task which depends on different factors that affect its quality. These factors that affect the quality of the research to the negative are limitations of the study. Accordingly, my research will face similar limitations. The first limitation which is particular to the usability and performance of my research work is that; there is no standard Amharic language corpus developed by Amharic language experts. Similarly, there is no big Amharic language corpus that provides suggestions and hence this affects the performance of the non-standard word spell checker and corrector. The second limitation of my study relates to time limitation in that time always goes forward and never come back. Research work requires huge time for quality work, but in most cases, time given for research work is limited and affects its quality. The third limitation that faces me in the course of conducting this research is lack of

literatures or similar works on this area that guides the organization of the study. The other limitation that faces the research work is finance. Financial resource is another factor that affects the quality of the research.

1.8 Methodology

1.8.1 Design science research method

Research methodology is a systematic way of solving a research problem . It is a science of studying how research is conducted scientifically. Researchers must employ suitable research design depending on the problem they intend to solve. For this research work, the researcher has employed design science research method. This is discussed in detail in chapter three, the methodology part.

1.8.2 Process Model

Design science research process model starts from identifying and defining problems and goes away through to communication. In the problem identification and motivation stage, the spelling errors occurred during typing Amharic language texts together with the absence of spell checker for Amharic are set as problems and motivate the researcher to engage in such research. The researcher then sets objectives with a view to model non standard word Amharic spell checker and corrector. Design and development of an artifact comes next. Demonstrating, how the application goes to solve the stated problem follows designing an artifact. Evaluation of the artifact designed and communication are the fifth and sixth stages respectively.

1.8.3 Modeling approach

Spell Checker systems can be designed using different approaches from the perspective of spell checker approach. The researcher uses Dictionary based minimum edit distance based approach for the error detector and corrector components respectively. These techniques are discussed in the subsequent chapters.

1.8.4 Tools

This thesis employs tools such as Python, NLTK, SPYDER, Tkinter, Notepad, PAGE, and ANACONDA to develop the prototype. Python is a very known in its power full natural language processing and artificial intelligence process. Spyder is used to run the python codes, PAGE used to design the user interface and to generate the python Tkinter.

1.9 Organization of the research

The thesis is presented in six chapters. The first chapter presents research background, problem description, the objective of the study, research questions, significance, scope and limitation of the study.

Chapter two presents various concepts relating to non-standard spell checker, the application of various algorithms and related works.

Chapter three discusses the methodology part.

Chapter four deals with system modeling for Non-standard Amharic word spell checker and corrector which includes; system architecture, techniques used for spelling error correction and Test and Prototype Evaluation Procedure.

Chapter five relates to experimental results and discussion. Finally, chapter six relates to conclusion and recommendation.

CHAPTER TWO

LITERATURE REVIEW AND RELATED WORKS

2.1 Introduction

Although Amharic is one of the maximum studied languages of Ethiopia, there's no consensus as to what number of POS there are for this language. Mersehazen (1935E.C) divided the phrase elegance into 8 categories. Noun, conjunction, preposition interjection, verb, adjective, pronoun, and adverb are examples of these [12]. Computers were used to clear up and automate complex problems in a extensive variety of sectors and fields in view that their debut.

Computational linguistics additionally referred to as herbal language processing (NLP) is a area of each laptop technology and linguistics that offers with the evaluation and processing of human languages the usage of virtual computers. NLP has additionally many programs such as; Automatic Summarization, Machine Translation, Speech Recognition (ASR), Optical Character Recognition (OCR), and Information Retrieval (IR).

Spell-checking is any other sizeable software of computational linguistics whose studies extends again to 196os. Today, spell checkers are critical thing of some of pc software program which include internet browsers, textual content processors and others. This bankruptcy devotes to provide the essence of spell checking, the styles of spelling errors, and the additives of spelling checking, mistakes detection and correction, strategies of mistakes detection and correction, and associated works.

2.2 The essence of spell checker

A spell-checker is computer software that identifies and, in many cases, corrects misspelled words in a document. It can be a standalone application or an add-on module integrated into an existing program such as a word processor or search engine. Spell checker provides error detection and correction functions [13]. While the “Error Detection” function verifies the validity of a word in the language the “Error Correction” suggests corrections for the misspelled word.

Spelling error correction can be either interactive or automatic [14]. In interactive spelling error correction, the spellchecker suggests more than one correction for each error and the user

has to select one for replacement on the other hand in automatic correction, the spellchecker decides the best correction and the error is automatically replaced with it. Automatic error correction is important for speech processing and Natural Language Processing (NLP) related systems where human intervention is not possible.

Fundamentally, a spellchecker consists of 3 components [8]. The first factor is blunders detector that detects misspelled words. The 2nd factor is candidate spelling generator that offers spelling recommendations for the detected errors, and the 0.33 factor is an blunders corrector that chooses the quality correction out of the listing of candidate spellings. Different or the same (merging in to at least one factor) strategies may be used to every factor.

2.3 Types of spelling errors

Spell checker techniques are designed on the basis of spelling errors. Studies have been conducted to analyze the types and trends of spelling errors. According to these studies spelling errors are generally divided into Typographic and Cognitive errors [15]. This sub section presents these classes of errors.

2.3.1 Typographic errors

Typographic errors are errors occurring when the correct spelling of the word is known but the word is mistyped mistakenly [16]. These errors are mostly related to the typing when a word is written incorrectly because a finger was placed on a wrong key of keyboard. In general, typographic errors are mainly caused due to keyboard adjacencies. According to a survey, 80 percent of typographic mistakes fall into one of the four categories below. [15].

A. Insertion error: is the addition of extra letter to a word. It occurs due to double pressing of a key or by accidentally hitting two adjacent keys while trying to hit one of them. Typing acress for cress is relevant example of insertion error. When we come to Amharic language; typing "ተ ሜ ፍ ች " for "ተ ሜ ዎ ች " typing "እ ያ ጠበኩ " for "እ ያ ጠኩ " are examples of insertion errors.

B. Deletion error: is omitting letter from a word. For example, typing acress for actress. Deletion error usually occurs when the eyes move faster than the hand. When we contextualize this to Amharic language; typing "ተ ሟ ች " for "ተ ሜ ዋ ች ", typing "እ ያ ኑ " for "እ ያ ጠኩ " are examples of deletion errors.

C.Substitution error: is the replacement of one letter by another letter in a word. This type of

error is mainly caused by replacement of a letter by some other letter whose key on the keyboard is adjacent to the originally intended letter‟s key. Typing acress for across is typical instance of substitution error. Contextualizing this to Amharic language; typing “እ ያ የ ኑ ” for

“እ ያ ጠኑ ” and typing “ተ ሜ ቆ ች ”for “ተ ሜ ዎ ች ” are examples of substitution errors.

D. Transposition error: is the interchange of letters positions in a word. Typing acress for caress is an instance of transposition error. When we come to Amharic language; typing “ተ ሞ ሪ ች ” for “ተ ሜ ዎ ች ”, typing “እ ያ ኑ ጠ” for “እ ያ ጠኑ ” are examples of transposition errors.

Literatures also provide other types of errors [17] such as; **Extra space:** when a word is split by an added space character for instance typing house instead of house, when we come to Amharic language typing “ተ ሞ ሪ ች ” instead of ተ ሞ ሪ ች . **Missing space:** when the space between consecutive words is missing for example typing “insidehome” instead of the “inside home”, when this error is contextualized in to Amharic language typing “መ ነ ገ ድ ላ ይ ” instead of “መ ነ ገ ድ ላ ይ”። **Replacing a letter by a space**; for example, typing “ho se” instead of “house” when contextualize to Amharic language typing “ተ ሪ ” instead of “ተ ሜ ” ። **Replacing a space by a letter;** for example, typing “theohouse” instead of the “house”, when we come to Amharic language typing “እ ሱ ቨበ ላ ” instead of “እ ሱ በ ላ ” and any combination of the above-mentioned errors.

2.3.2 Cognitive (real word) errors

Cognitive errors occur when writer does not know or has forgotten the correct spelling of a word or the word is correct but does not fit to the context of the sentences [18]. It is assumed that in the case of cognitive errors, the pronunciation of misspelled word is the same or similar to the pronunciation of intended correct word. Typing three for tree, there for their are relevant examples of cognitive errors.

When we come to Amharic language; real word errors can be understood from the following sentence; “ከ ተ ወሰ ነ ቀ ና ት በ ኋ ላ የ 12ኛ ክ ፍ ል ፈ ተ ና ይ ሰ ሜ ። ” The word “ይ ሰ ሜ ” is meaningfulword but contextually it is not the intended word to the sentences. The correct word that fits the context is “ይ ሰ ጣ ”. Taking the context in to consideration the correct sentences is;“ከ ተ ወሰ ነ ቀ ና ት በ ኋላ የ 12ኛ ክ ፍ ል ፈ ተ ና ይ ሰ ጣ ። ”

In both cases (typographic and cognitive errors), the issue is to detect the word error and either suggests correct alternatives or automatically replace it with the appropriate valid word [19]. The detection of cognitive errors needs higher level knowledge compared to the detection of typographic errors. For cognitive (real word errors), it is often not possible to separate the problem of error detection from that of correction [19].

2.3.3 Non-Standard word errors

The word non-standard words in language means list of words which are derived from other languages and some known abbreviations. Non-standard words commonly used in Amharic language. For example "የ ዩ ኤን መሪ ዎች በ መዲና ዋ የ አ ምስ ት ቀን ስ ብሰ ባ አ ድር ገ ዋ ል ።" has one non- standard word which may be not in most Amharic dictionary corpus.

2. 4 Techniques of spell checker

A spell checker is consisted of mainly two components, a Lexicon (Dictionary) and cluster of techniques that use this lexicon for spell checking [13]. These techniques generally provide error detection, error correction and ranking of correction functions. In the sub sections below, a discussion of techniques that provide error detection and correction will be made.

2.4.1 Error detection techniques

Candidate words are a series of characters separated by a space bar or punctuation marks [20]. If a candidate word has a meaning, it is legitimate; otherwise, it is a non-word. Checking if an input string is a valid index or dictionary term is generally the first step in the error detection process. N-gram analysis and dictionary lookup are the two most well-known methods for detecting errors. Text recognition systems rely on n-gram algorithms, whereas spellcheckers mostly on dictionary lookup [20].

A. N-gram Analysis

N-gram analysis is one of the popular methods for detecting spelling errors [4]. Normally, it is used to detect errors made by Optical Character Recognizers (OCR) devices. N grams are n letters subsequences of words or strings. N stands for one, two or three. One letter n-grams are referred to as unigrams or monograms; two-letter n-grams are referred to as bigrams; and three- letter n-grams are seen as trigrams.

In order to pre-compile an n-gram table, a dictionary or corpus of text is usually required. The table stores n-gram's existence or frequencies, any n-grams in an input string that have

nonexistence or low frequencies are classified as probable misspellings. The table has a variety of forms, such as binary bigram array or binary trigram array. In the case of binary bigram array, it has two-dimensional array of size 26 * 26 whose elements represent all possible bigrams. For each element, if it occurs in at least one word (string) in predefined dictionary or text, then its value set to 1, otherwise set to 0. A binary trigram array could have three-dimensional array. Since these binary n-grams do not indicate the position of each n-gram within each word, they are non-positional binary n-gram arrays. Moreover, it is said that a set of positional binary n- gram arrays are able to detect error more accurately. Because each element in the positional binary n-gram arrays matches the exact position within each word. However, this raises the storage space problem due to the large capacity of the positional arrays. Since most misspellings do not contain any impossible n-grams, so n-gram analysis techniques are not good at detecting human generated errors but good at detecting machine-generated errors.[4]

B. Dictionary look up

A dictionary is a list of words that are assumed to be correct [14]. Dictionaries are represented in many ways, each with their own characteristics like speed and storage requirements. Large dictionary might be a dictionary with most common word combined with a set of additional dictionaries for specific topics such as computer science or economy. Big dictionary also uses more space and may take longer time to search. The Non-standard words can be detected as mentioned above by checking each word against a dictionary. The drawbacks of this method are difficulties in keeping such a dictionary up to date, and sufficiently extensive to cover all the words in a text. At the same time one should keep down system response time. Dictionary lookup and construction techniques must be tailored according to the purpose of the dictionary. Too small a dictionary can give the user too many false rejections of valid words, too large it can accept a high number of valid low frequency words. Hash tables are the most common used technique to gain fast access to a dictionary. To search up a string, compute its hash address and obtain the word stored at that address in the hash table that has already been built. A typo is detected if the word stored at the hash address differs from the input string.. Hash tables main advantage is their random-access nature that eliminated the large number of comparisons needed to search the dictionary. The main disadvantage is the need to invent a clever hash function that avoids collisions. To store a word in the dictionary we calculate each

hash function for the word and set the vector entries corresponding to the calculated values to true. To find out if a word belongs to the dictionary, we calculate the hash values for that word and look in the vector. If all entries corresponding to the values are true, then the word belongs to the dictionary, otherwise it does not.

2.4.2 Error correction techniques

Spell correcting refers to the attempt to endow spell checkers with the ability to correct detected errors, i.e. to find the subset of dictionary or lexical entries that are similar to the misspelling in some way. Error correction consists of two steps; the generation of candidate corrections and the ranking of candidate corrections [14]. To find one or more viable correction words, the candidate generation procedure commonly uses a precompiled table of permissible n-grams. To rank order the candidates, the ranking procedure commonly uses a lexical similarity measure between the misspelled text and the candidates or a probabilistic assessment of the likelihood of the repair. Most of the time, these two procedures are viewed as independent processes and carried out in order. However, some techniques can skip the second step, leaving the user to rank and choose the best option.This thesis is confined to Non-standard error, and hence this sub section confines its discussion to the methods used to correct non real word errors as follows;

A. Minimum Edit Distance

Minimum edit distance technique is the most studied and used technique for spelling correction to date. Minimum edit distance as its name suggests is a technique where the minimum number of editing operations (insertions, deletions, substitutions, and transpositions) required transforming one string into another [20]. The term, minimum edit distance between two spellings, say w_1 and w_2, refers to the smallest number of editing operations that need to take place in order to transform w_1 to w_2. The editing operations referred to here are insertions, deletions, substitutions, and transpositions. Normally, one is concerned about the minimum edit distance between misspelled words in a text to a word in the dictionary. The large majority of spelling errors could be corrected by the insertion, deletion or substitution operation of a single letter, or the transposition of two letters [17]. If a misspelling can be transformed into a dictionary word by reversing one of the error operations (i.e. insertion, deletion, substitution, and transposition), the dictionary word is said to be a plausible correction [21]. This technique

is limited to single-word errors. Thus, the number of possible comparisons is greatly reduced. Edit distance algorithm detects spelling error by matching words of four to six characters in length to a list of words with high frequency of occurrence [17]. If the search word is not found in the list, the word is looked up in a dictionary in which words have been sorted according to alphabetical order, word length, and occurrence of characters. If the search word cannot be found in the dictionary the correctly spelled word is searched by the algorithm. This search is performed on both word level and character level. This is to say, under the `one error' assumption, all words that differ in length by one character or differ in the occurrence of characters by less than or equal to two-bit positions are checked against the detected words using the spelling rules. All the remaining words are thus bypassed.

Where a word in the dictionary is one character longer than the detected word, then the first character in the dictionary word that is different is discarded and the rest of the characters are shifted one-bit position left [20]. If the two words match, then the word in the dictionary is reported to be the correctly spelled word by a single insertion. For example, the word apple is detected as a misspelled word and apple is a word in the dictionary and they differ by one character. The character `p' is discarded from apple and the rest of the characters are shifted left. Apple is then compared with the detected word and a match is found. Therefore, apple is reported as the correct spelling for apple [20].

On the other hand, if the word in the dictionary is one character shorter, then the first character in the detected word that is different from the matching character in the dictionary word is considered to be incorrect [20]. Thus, that particular character in the detected word is discarded and the rest of the characters in the misspelled word are shifted one-bit position left. If there is a match for the new misspelled word and the dictionary entry, the word in the dictionary is reported to be the correctly spelled word by a single deletion. For example, the word “apple” is detected as a misspelled word and apple is a word in the dictionary and they differ by one character. The character `l' is discarded from “apple” and the rest of the characters are shifted left. apple is then compared with the word in the dictionary and a match is found. Therefore, apple is reported as the correct spelling for apple [20].

Where the lengths of the word in the dictionary and the misspelled word are the same, but they

differ by one-character position, then the dictionary entry is reported as a candidate correction as they differ by a single substitution. For example, the word “aplle” is detected as a misspelled word and apple is a word in the dictionary and they differ by one character. The first `l' is replaced by `p'. The resultant word is then compared with the dictionary word and a match is found. Therefore, apple is reported as the correctly spelled word for “aplle”.

Where the lengths of the word in the dictionary and the misspelled word are the same, but they differ in two adjacent positions, the characters are proposed to be swapped. If the two words are the same, there is a match by a single transposition [20]. For example, the word “apple” is detected as a misspelled word and apple is a word in the dictionary and they differ by one character. The positions of the characters `l' and `p' are swopped. The resultant word is then compared with the dictionary word and a match is found. Therefore, apple is reported as the correctly spelled word for “apple”.

B. Similarity Keys

The essence of similarity key techniques is the mapping of every word into a key [20]. The mapping is chosen so that similarly spelled words will either have similar or identical keys. When a key is computed for a misspelled word, it will provide a pointer to all similarly spelled words in a dictionary, and these dictionary entries will be returned as candidate corrections. This is to say, all words in a dictionary having similar key values compared to the key of the current misspelled word, will be returned as possible correct words. Due to the fact that it is not necessary to compare the misspelled word with every dictionary entry, similarity key techniques are fast.

Similarity key techniques are based on transforming words into similarity keys that reflect the relations between the characters of the words, such as positional similarity, material similarity, and ordinal similarity [22].

Positional similarity: as indicated by its name, it refers to the extent to which the matching characters in two strings are in the same position. It generally appears in an OCR text and literary comparison and is said to be too restricted to be used on its own for spelling correction[16].

Material similarity: refers to the extent to which two strings consist of exactly the same characters but in different order. Correlation coefficients between the two strings (i.e. the misspelled word and a word that consists of the exact same characters but in different order)

have been used as a measure of material similarity [5]. Material similarity is seen as not precise enough for the spelling correction task as all anagrams are materially similar. For example, the word angle is an anagram of the word gleans and they are materially similar.

Ordinal similarity: Similar to position similarity, ordinal similarity indicates the extent to which the matching characters in two strings is in the same order.

Each dictionary word is given a key, and only dictionary keys are compared to the key generated for the non-word. The Non-standard word for which the keys were calculated. As a recommendation, the word with the most comparable keys is chosen. This method is fast because it only processes words that have similar keys. With a good transformation algorithm, this method can handle keyboard errors.

C. Rule based Techniques

Rule-based techniques involve algorithms that attempt to represent knowledge of common spelling error patterns, for transforming misspelled words into correct ones. The knowledge is presented as rules [20]. They function by applying a set of criteria to the misspelled word that captures frequent spelling and typographical problems. These principles appear to be the "inverses" of frequent mistakes. This algorithm generates a correction suggestion for every correct word it generates. The rules include probabilities as well, allowing you to rank the ideas by adding up the probability for the applicable rules. Edit distance may be thought of as a specific instance of a rule-based system with a restriction on the number of rules that can be used.

D. N-gram Based Techniques

An n-gram is a subsequence of a sequence of letters or words with n = 1, 2, or 3 letters. If n = 1, 2, or 3, a unigram, bigram, or trigram is referred to, correspondingly.In other words, n-grams are substrings of length n. N-grams can be used in two ways, either without a dictionary or together with a dictionary. Spell correctors employing n-gram-based techniques follow three processes: error detection; candidate suggestion; and rank similarity [20].

Rank Detection

N-grams have been widely used in correcting misspellings in an OCR text in order to capture the lexical syntax of a dictionary and suggest valid corrections. A dictionary is partitioned into

subsets according to word lengths. Each subset has positional binary n-gram matrices and these matrices captured the structure of strings in the dictionary. Each output word from the OCR device could be checked for errors by verifying if all its n-grams have value 1.

- If at least one binary n-gram of a word has value 0, it is indicated to have a single error.
- If more than one binary n-gram of words have value 0, the positions of the errors are then indicated by a matrix index that is shared by the n-grams with 0 value. The rows or columns in the matrices specified by the common index of the erroneous n-grams are then logically intersected in order to find possible suggestions.
- If the result of the intersection indicates that only one n-gram has value 1, a candidate correction is found.
- If more than one n-gram suggestions are found, the checked word is rejected.

The advantage of this technique is that it prevents an exhaustive dictionary search. However, it runs a risk of resulting in a Non-standard as a correction. This technique only handles substitution errors.

Candidate Suggestion

The second spell correcting procedure is to suggest candidate corrections. N-grams have often been used as access keys into a dictionary for locating possible suggestions and as lexical features for computing similarity measures. The number of the non-positional binary trigrams that occurred in both a misspelled word and a dictionary word were computed. Non-positional n- grams were referred to as n-gram arrays that did not indicate the positions of the n-grams within a word.

The lexical features in this case were trigrams. The similarity measure was then computed by a function called Dice coefficient which was

$$D(n_m; n_d) = 2(c / (n_m + n_d));$$

Where

- c is the number of shared/common trigrams for both the misspelled word and the word in the dictionary,
- n_m is the length of the misspelled word, and n_d is the length of the dictionary word.

Note that n_m and n_d can be interchanged. Furthermore, the trigrams of the misspelled word are used as access keys. These trigrams were used to retrieve words in the dictionary that have

at least one trigram that is in common with the misspelled word. The drawback of this particular technique is that any words shorter than three characters cannot be accurately detected because one single error can leave no valid trigrams intact. For misspellings containing more than one error, the function for similarity measure was changed to

$$D(n_m; n_d) = 2(c / \max(n_m + n_d));$$

Where max (n_m, n_d) is the highest probability of the common trigrams.

Rank Similarity

The third spell correcting procedure, similarity ranking, is not executed in any of the example schemes discussed to date. However, n-gram based techniques can also be used to find and rank candidate suggestions. Both misspelled and correct words are represented as vectors of lexical features (with unigrams, bigrams, and trigrams as possible candidates for the features of the lexical space) to which conventional vector distance measures can be applied. The measures then form the basis for ranking candidate suggestions.

These techniques first position each dictionary word at some point in an n-dimensional lexical feature space in where words are represented as n-gram vectors. The dimension of a lexical-feature space n can be very large. For example, if the lexicon consists of 10,000 words, one may use trigrams (sequences of three consecutive letters) as the feature, then n = $10^1$2 (10^{4x3}).

E. Probabilistic techniques

They are, simply put, based on some statistical features of the language [23]. Two common methods are transition probabilities and confusion probabilities. Transition probabilities are similar to n-grams. They give us the probability that a given letter or sequence of letters is followed by an16 another given letter. Transition probabilities are not very useful when we have access to a dictionary or index. When given a phrase to correct, the system decomposes each string into letter n-grams and searches the lexicon for word suggestions by comparing string n-grams to lexicon-entry n-grams. Given character confusion probabilities, the recovered candidates are sorted by the conditional likelihood of matches with the string. Finally, the optimal scoring word sequence for the phrase is determined using a word-bigram model and a specific method.They claim that the system can correct Non-standard errors as well as real word errors and achieves a 60.2 % error reduction rate for real OCR text.

F. Neural Networks

Neural networks are also an interesting and promising technique, but it seems like it has to mature a bit more before it can be used generally [23]. Back-propagation networks are now used, with one output node for each word in the dictionary and an input node for every conceivable n-gram at every position of the word, where n is commonly one or two. Only one of the outputs should be active at any one time, showing which dictionary terms the network recommends as a correction. This strategy works for tiny dictionaries (under 1000 words), but it does not scale well. The time requirements are too big on traditional hardware, especially in the learning phase.

2.5 Usability

Human activities implicitly articulate needs, preference and design vision. Artifacts are created in answer to a need, but they always do more than that throughout their acceptance and appropriation; each design opens up new possibilities for action and interaction.Ultimately, this activity articulates further human needs, preference and design vision due human interaction with computer by nature it is dynamic every time travel with change but all these things should follow the standards to have acceptance by all users for this propose usability standard is one of the very common tool in producing right products this usability standard mainly focus on four measure issues these are ease of use, efficiency, effectiveness, memo ability and satisfaction. In developing Spelling checker system, ISO standard 9241 have highly paramount in validating the system and meet real client requirement to accomplish and facilitate the E-learning delivery system without any delay and bugs.

2.5.1 ISO 9241 part 1-part 17

I eneral Requirement Environment

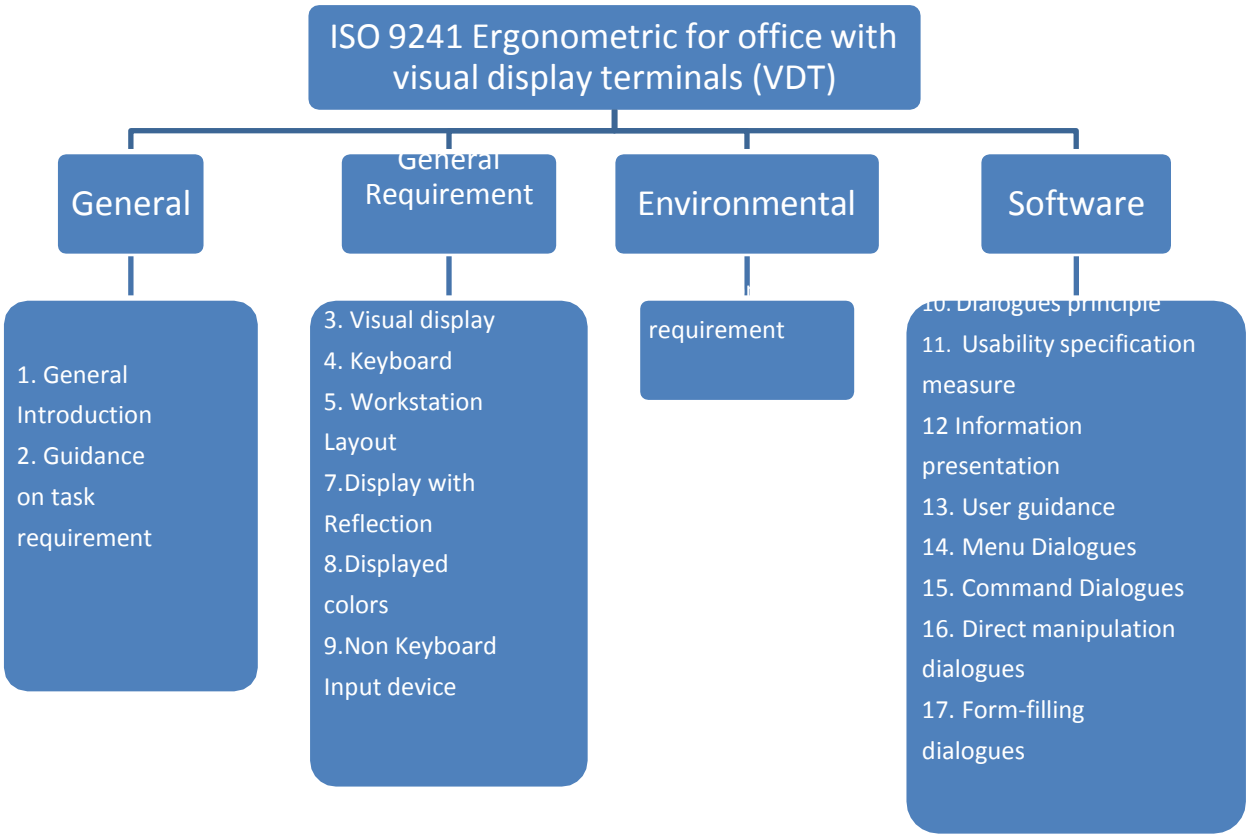

Figure 1 ISO 9241 part 1-part 16 adopted from [24]

Isometric is best technique for software evaluation tool. It is used to measure the usability of software products in line with ISO 9241 part 11 standards. Isometrics is user-oriented. The most relevant part of ISO 9241 in the present context is Part 11, entitled Usability specification measure. It is a part which deals exclusively with software aspects. ISO 9241 Part 11 criteria are listed below [24].

2.5.2. ISO 9241 Standards

Theoretical frame work for ISO 9241 usability measure

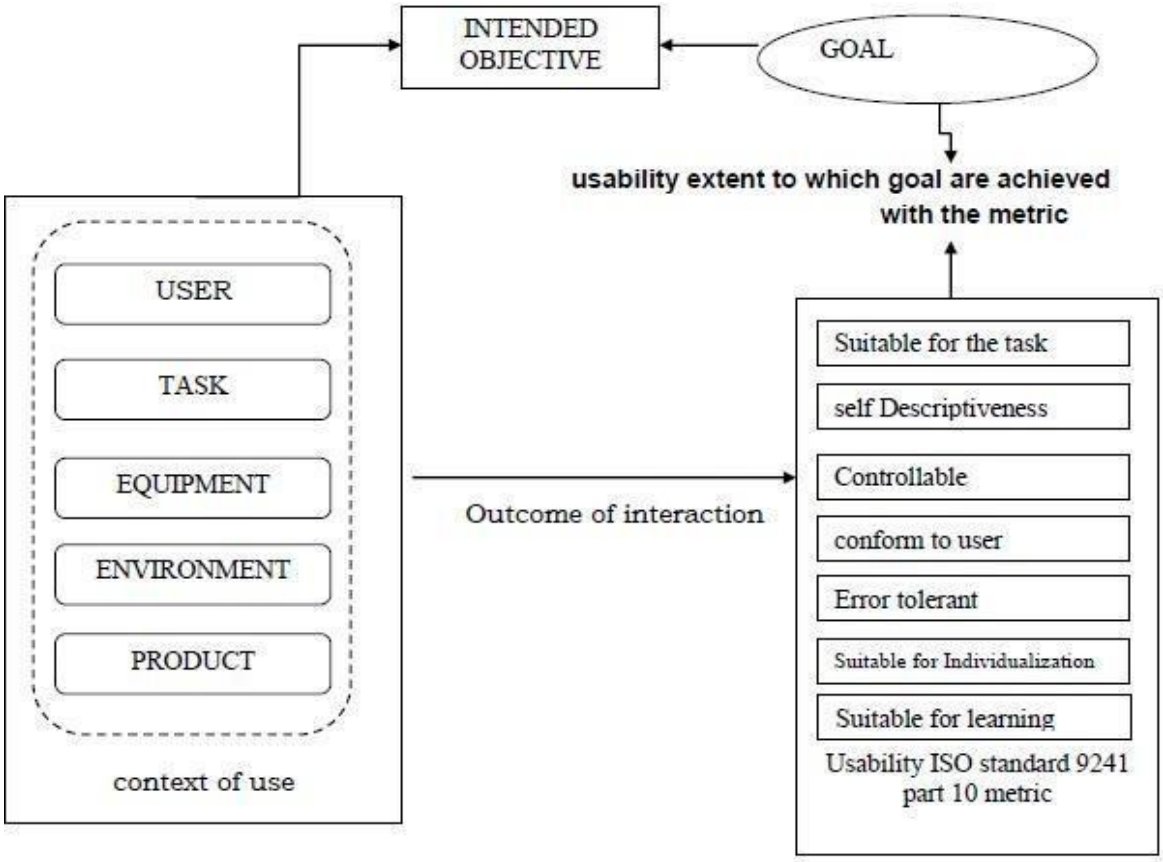

Figure 2 Theoretical frame work for ISO 9241 usability measure

ISO 9241 core points are concerned with usability of hardware, software and environment attributes. To mention some of the sub Parts 3 to 9 relate to hardware design requirements and guidelines. Parts 10 to 17 focus with software attributes [24]. The seventeen parts of ISO9241 list on the following figure.

Suitability for the task a discussion is suitable for the task, if it supports the user in the effective and efficient completion of the task. The discussion presents the user with only those concepts that are related to the user's task. Self-descriptiveness a dialogue is self-descriptive, if each dialogue step is straight away comprehensible through feedback from the system or is clarified to the client on his or her requesting the relevant information. Controllability A

dialogue is controllable, if the user is capable to maintain track over the whole course of the interface until the point at which the goal has been met. Conformity with user expectations A dialogue conforms with user expectations if it corresponds to the user's task education, knowledge, experience, and to usually held conventions. Error tolerance A dialogue supports error tolerance if, despite evident errors in input, the future result may be realized with either no or minimal corrective action having to be in use. Suitability for individualization A dialogue is suitable for individualization, if the dialogue system is constructed to allow for modification according to the user's individual needs and skills for a given task [24].

Suitability for learning Dialogue is suitable for learning, if it guides the user through the learning stages minimizing the learning time. These principles should be applied to the design and evaluation of every dialogue. However, their varying relevance in different areas of application or different dialogue techniques has to be taken into account. ISO 9241 gives the following examples for situations which may cause different emphasis among the design principles:

- The aims of the organization of which the user is a part
- The requirements of the user group
- Different tasks which are to be supported by the application, available technologies and resources.

Therefore, it is necessary to take these aspects into account before or during the evaluation of a software product, so that the sampled data can be correctly interpreted in the given context [24]. On the basis of ISO 9241 Part ten, ISO 9241 Part 10 formulates seven essential principles in linking the design and assessment of the conversation, aiming at a user-oriented approach in software evaluation. The summative versions of Isometrics exhibit great dependability of its subscales and extract accurate information regarding variations in software system usability.

2.6 Review of related works

1. Wordlist and Spell checking for Amharic and Tigrigna By Biniam Gebremichael April 2011. Develop Amharic and Tigrigna Spelling checker in open office desktop application, to help Geez Natural Language Processing developers. This of course requires installing an open office itself and a language-specific dictionary. This application is not research-based instead project-based and in developing this prototype

he uses a corpus of words.

2. **Design and Implementation Of Morphology Based Spell Checker** [25] this study is a desktop application. The system Afaan Oromo spell checker is developed via Microsoft Visual C# 2010 and the study uses for error detection dictionary lookup for correction Morphological analyzer. and lastly, the study scores Lexical Recal 88.62%.
3. **Design and implementation of Online Punjabi Spell Checker based on Dynamic Programming** [26] In this study the Spell Checker has three mechanisms: An error detector that notifies misspelled words, a candidate spelling initiator that gives as output spelling suggestions for the detected misspell string and an error corrector that chooses the finest correct spellings. the study uses tools ASP.NET, a language with SQL Server 2005. The study uses the technique dynamic approach (Top-down dynamic approach and Bottom-up dynamic approach.) of the list of candidate spelling. According to the study work for Punjabi words, the System gives the result accuracy of 80%.
4. **Improved Spelling Error Detection and Correction for Arabic** [27]**:** By Mohammed Attia, 2012. A dictionary, an error model, and a language model are the three key components of this research. A morphological transducer and a big corpus are used in this work to analyze a lexicon of 9.3 million Arabic words. By studying error kinds and implementing an edit distance-based re-ranker, the error model is improved. Our method outperforms Microsoft Word 2010, Open Office Ayaspell, and Google Docs by a large margin.
5. **Checking techniques in NLPA Survey Neha** [28] In this study the researcher explains the various techniques for spell checking and error correction. This study also provides information about various available spell-checking systems developed for different Indian languages. In this paper, two techniques for spell checking are described. These are N-Gram Analysis based on statistical technique and Dictionary lookups Finally the researcher score accuracy 85%.

In this chapter the researcher review, different works which are done on spelling checker for different languages and the technique they are using. Almost all of the local reviews are not research based they are project based and desktop applications. The Design and implementation Of morphology based Spell checker for Afan Oromo uses two technique for

detecting error dictionary lookup and for correction morphological analyzer and his recall is 88.62%. Having a larger corpus of data presents information about how frequently words are used in a document. reviewing many documents used to check whether there are papers related directly to our work so that it is possible to apply or integrate the necessary components of the work.

By comparing all the techniques the researcher decides to use dictionary lookup for detecting the errors, Edit distance for making suggestions, and frequency of words for Ranking of the relevant words development of the study model.

According to the findings, there is no paper focused on the Amharic non-standard Spelling checker and corrector. Therefore, it is necessary to research systems modeling for the Amharic spelling checker and correction.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Introduction

This chapter presents the research methodology employed by the researcher. Research should follow a valid research methodology so that it be accepted and formalized internationally. Research methodology is a systematic way of solving a research problem [29]. It is the science of studying how research is conducted scientifically. Its main aim is to provide the work plan of research. To accomplish the research objectives, the research should follow an appropriate research methodology. Researchers must employ a suitable research design depending on the problem they intend to solve. Research methodology differs from research to research depending on the problem the researcher wants to address and the needed outcome.

Under this chapter, the researcher presents the methodology the research employs; the process model of the research, modeling approach (techniques of the research), developmental tools and techniques, and the evaluation and testing procedures.

3.2 Research Method

The researcher has employed the design science research method. Design science research is a kind of research method which produces a viable artifact in the form of a construct, architecture, model, and method [30]. Its objective is to develop technology-based solutions to a problem. This study wants to model Non-standard word error spell checker Amharic language and hence it is best addressed through design science research methodology. The basic tenet of design science research is that knowledge and understanding of a design challenge and its solution are gained via the construction of an artifact. The construction of an inventive, meaningful artifact for a specific issue domain is a requirement of design science research. Because the artifact is intended to solve a specific problem, it must be useful. The design science research method consists of the environment which defines the problem space in which the phenomena of interest reside. The environment is composed of people, organizations, and technologies.

Design science addresses research problems through developing and building theories, and artifacts to justify or evaluate the business need with an assessment of research activities. The knowledge base provides the raw materials from and through which research is accomplished.

The foundations and techniques make up the knowledge base. Prior Design science research and findings from reference disciplines offer the basic theories, frameworks, instruments, structures, models, methodologies, and instantiations that are employed in the research study's development and construction stages. Methodologies are used to justify and assess a certain phase. Rigor is attained by the proper application of established foundations and procedures. In design science, computational and mathematical methods are primarily used to evaluate the quality and effectiveness of artifacts. In order to achieve the objectives of this study, the researcher finds design-science research methodology best and uses it.

Conceptual Framework for Non-standard error Amharic language spelling checker and corrector

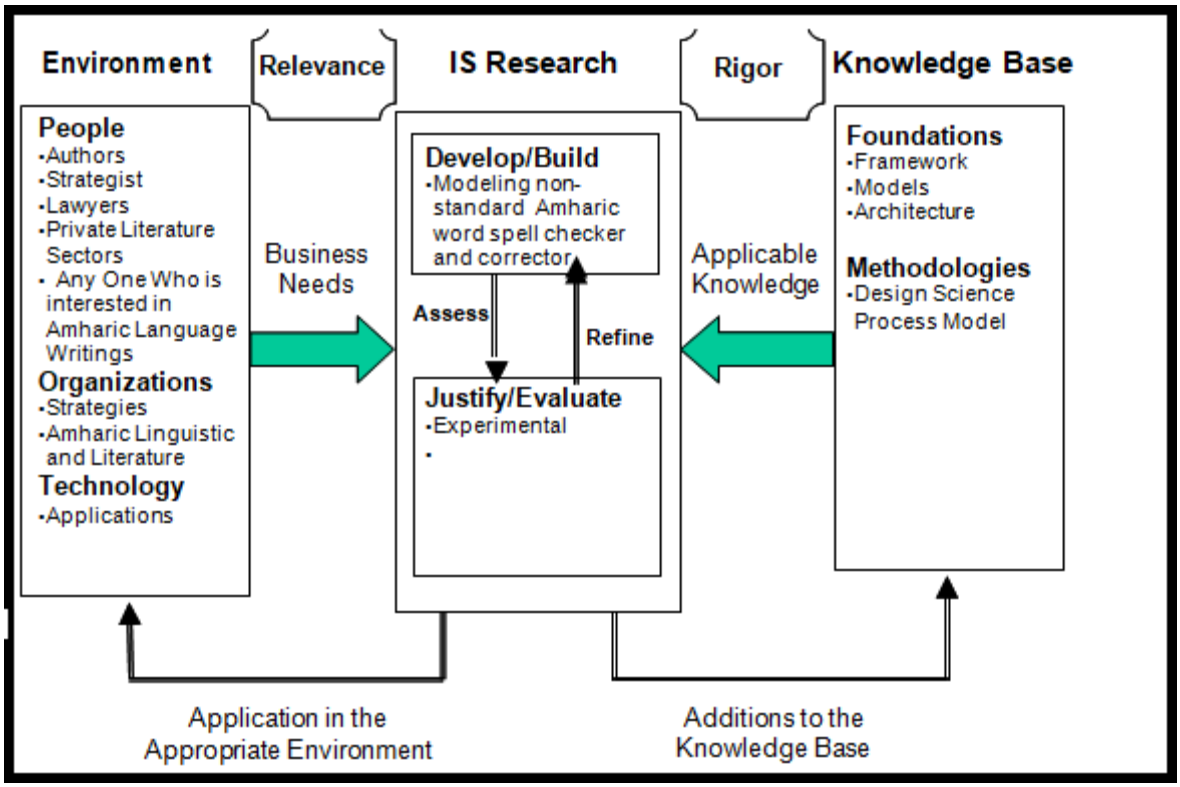

Figure 3 Design Science Research Methodology Adapted from [31]

This study has explored introducing and building an artifact in Non-standard error Amharic language spell checker and corrector. The environment of this study is on the Amharic words especially Non-standard errors that occurred during typing the words.

Relevance of the research is conducted. This is because Non-standard word error Amharic

language spell checker and corrector simplifies the workload of many sectors to check and edit the spelling errors written in Amharic and saves time for editing. So far, to the knowledge of the researcher, there is no application (software) that runs on a computer desktop. Due to this, the researcher understands that errors occur in many cases and sectors. For example, a simple example is in different mass media like Television and social media, it is observed that many spelling errors written in Amharic are displayed. These errors together with the absence of Non-standard error Amharic Language spell checker and corrector motivate the researcher to search on [31].

The researcher has reviewed papers in the area of NLP especially on spell checker systems to have a better understanding of the problem area and to avoid redundancy of research works. An experiment is conducted to determine how well an artifact works with appropriate words. The knowledge base of design science plays a great role in providing materials that help to design the Non-standard error Amharic spell checker and corrector. This is because; the knowledge base of design science contains the foundation like theories, frameworks, models, and selecting suitable methodologies [31].

3.3 Process Model

Design science research process model starts from identifying and defining problems and goes away through to communication. The stages, process model goes through are illustrated below in the figure as well. In the problem identification and motivation stage, the spelling errors that occurred during typing Amharic language texts are set as problems and motivate the researcher to engage in such research. Here, the problems identified are the spelling errors and the absence of an Amharic language spell checker and corrector. The problems identified motivate the researcher to look here and there, and to come up with solutions. After identifying and defining the problem, the researcher defined objectives to model and implement a Non-standard word error Amharic language spell checker and corrector. This objective supports people who write texts through the Amharic language. The design and development of an artifact is the third stage. At this stage, the researcher practically deploys the standalone software developed for any operating system by configuring all tasks and methods with all required components. Demonstrating comes next to designing an artifact. An attempt has been made to show how the application (software) goes to solve the stated problem with all its

strength and easiness. Here different experimentations have been made to demonstrate. In the fifth stage, an evaluation of the artifact is made to observe and measure how well the designed artifact supports users in detecting and suggesting words for Non-standard errors using the ISO standard of 9241 usability engineering. In the last stage of communication, the researcher presents the work and sets ways for publishing the paper to contribute to such work after approval of the board of examination. See figure 4.

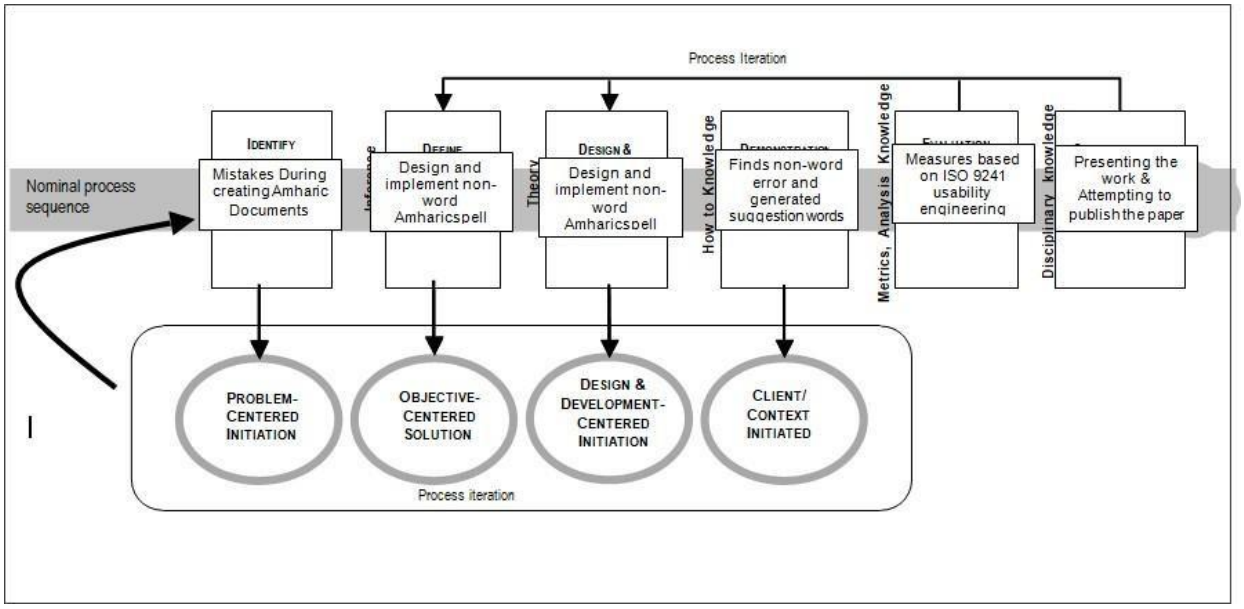

Figure 4 Design Science Research Methodology (DSRM) Process Model adapted from [32]

3.4 Modeling Approach

Spell Checker systems can be designed using different approaches from the spellchecker approaches as discussed in chapter two. The researcher uses a Dictionary-based minimum edit distance-based approach for the error detector and corrector components. The valid words stored in a dictionary are cross-checked to the words typed by the users. The words not found in the dictionary are considered misspelled and sent to the error corrector component. The reason behind employing minimum edit distance is to make use of its simplicity. In addition minimum edit distance is prefere to work with non English words.

3.5 Developmental Tools

Python is a high-level object-oriented programming language, and it is simple and easy to learn its syntax readability and reduce the cost of maintenance. Python contains a lot of packages using this one can do code reusability. Natural language processing with Python can

extract information from unstructured text, either to guess the topics or identify named entities. Using Python, through parsing and semantic one can analyses language structure, integrate systems drawn from various languages and artificial intelligence. Python is an excellent language for natural language processing. It is a scientific language and excellent for natural language processing [33]. In addition to the above reason python NLTK has thousands of library regarding a natural language processing.

3.5.1 Python IDEs and Python Text Editors

IDLE is the official Python cross-platform IDE, written using Tkinter. The researcher uses Spyder. SPYDER Python IDE is one of the most used Python IDEs. Scientific Python Development Environment (SPYDER) is an abbreviation for Scientific Python Development Environment. The Scientific Python Community is the primary user of this IDE.Tools and libraries like Numpy, Scipy, Matplotlib, etc are built with this python IDE. There are also some features a SPYDER makes the best IDE like its Syntax coloring and breakpoints, Code auto-complete and variable explorer, etc.

Python Code Editors unlike IDE are just simply programs that allow you to write code. The researcher uses Visual studio (VS) code. VS Code was developed by Microsoft and released in the year 2015. VS code editor supports Python snippets, syntax highlighting, brace matching, and code folding.

3.5.2 MySQL Database and Notepad ++ As Dictionary

MySQL is a relational database management system based on Structured Query Language (SQL). The application is used for a wide range of purposes, including data warehousing, e-commerce, and logging applications. Notepad ++, on the other hand, is a text and source code editor for Microsoft Windows. Tap editing, syntax highlighting, code folding, scripting, and markup languages are all supported. The researcher can use both the above tools as a dictionary or as a database for holding a large set of Amharic words. But to make use of its easiness and zero configurations, the researcher decides to use Notepad ++ as a dictionary.

3.5.3 Natural Language Tool Kit (NLTK)

Striping, part of speech tagging, stemming, sentiment analysis, topic segmentation, and named entity identification are among the most frequent algorithms in NLTK.. NLTK helps the computer to analyze, preprocess, and understand the written text. The researcher uses NLTK

for striping characters.

3.5.4 Tk, Tcl and Page

The researcher uses PAGE as Graphical User Interface (GUI) builder. PAGE is a python and Tkinter drag-and-drop GUI generator that creates python modules that show a reasonably basic GUI built from Tk and tkk widget sets utilizing the location, Geometry Manager. A page is a cross-platform tool running on any operating system which has Tcl/Tk installed. PAGE output requires only Python and Tkinter and runs on Linux, UNIX, Windows, OSX, and even Rasperian.

3.6 Evaluation and Testing Procedures

Performance measurement is one of the objectives of the spellchecker system, thus measuring the performance of the designed model is among the tasks of this work. This research uses both prediction accuracy formulas and the user satisfaction through a questioner.

The research uses a prediction accuracy, lexical recall and lexical precision and accuracy as a measurement of the model.

Thus, the evaluation of the designed model is also done by different users who are familiar with Amharic language writings. A questionnaire is prepared based on ISO 9241 standard usability engineering and distributed to respondents. The performance of the prototype is analyzed based on tables and percentages. It also uses qualitative method analysis of the descriptive one.

CHAPTER FOUR
SYSTEMS MODELING FOR NON-STANDARD AMHARIC WORDS SPELLING CHECKER AND CORRECTOR

The main objective of the research is to validate, suggest and correct the non-standard word errors that occurred during a typing Amharic language word. The researcher proposed a standalone application which is a desktop-based application. The reason the researcher chooses a standalone desktop application is that most of the time the need is more in the office of different sectors like schools, bureau of justice, different Woredas offices, hospitals, statistics and strategy offices, etc. In those offices, the secretaries write an average of 20-30 pages per day.

4.1 Architecture of the System

4.1.1. Tokenizer component (Case-Fold)

As shown in Fig 5, this component split a block of text into single words, digits, and punctuation marks. In Amharic Language, like in English languages, the blank space shows the end of one word. Furthermore, sentence boundaries and punctuations are almost similar to the English language except (i.e. a sentence may end with a period (: :), line break, a question mark (?), or an exclamation point). Thus, space marks are used as the explicit delimiters or token separators. Every time space is encountered, the word after the space becomes a token. The output of this component (i.e. list of tokens) becomes an input to the error detection module.

4.1.2. Error Detection component

The error detection component is responsible for checking whether the input word is misspelled or not. The error detection component works first by looking at the input word in the Amharic word dictionary. If the input word exists in the root word dictionary, the spell checker does nothing. Otherwise, the input word will be sent to the Error Checker for further processing. The Error Checker component takes the words from the Tokenizer component and checks if the words are found in the Amharic dictionary one by one.

Finally, to determine if this word is acceptable, the class of this root word is checked in the root word dictionary. If it is found in the dictionary, the system will recognize it as a valid word (i.e.

no further processing is needed), otherwise, the error detection component will recognize it as a misspelled word. Since python by default uses a hash array, we have already adopted it. Hashing is a well-known and efficient lookup strategy. If the word stored at the hash address is the same as the input string, there is a match. However, if the input word and the retrieved word are not the same or the word stored at the hash address is null, the input word is indicated as a misspelling. The random-access nature of hash array eliminates the large number of comparisons required for lookups.

Input: words from Tokenizer component
Output: list of Error words or Correct Flag
Start

Step 1. Read words from dictionary
Step 2. Strip each word in dictionary and append them to a list
Step 3. Accept Block of Text from user
Step 4. Fold and strip the block of text and append them to List of words
Step 5. If the word in step 4 is found in List of Words in dictionary flag the word as **CORRECT**
Step 6. Else If the word in step 4 is not found in List of Words found in dictionary
a. Send the word to **ERROR CORRECTOR COMPONENT**

END

Algorithm 1 Algorithm for Error Detection

4.1.3. Error Corrector component

This is the main component that returns with some suggestion words by taking the misspelled word and the block of words from the dictionary. This component makes a probable suggestion to the misspelled words.

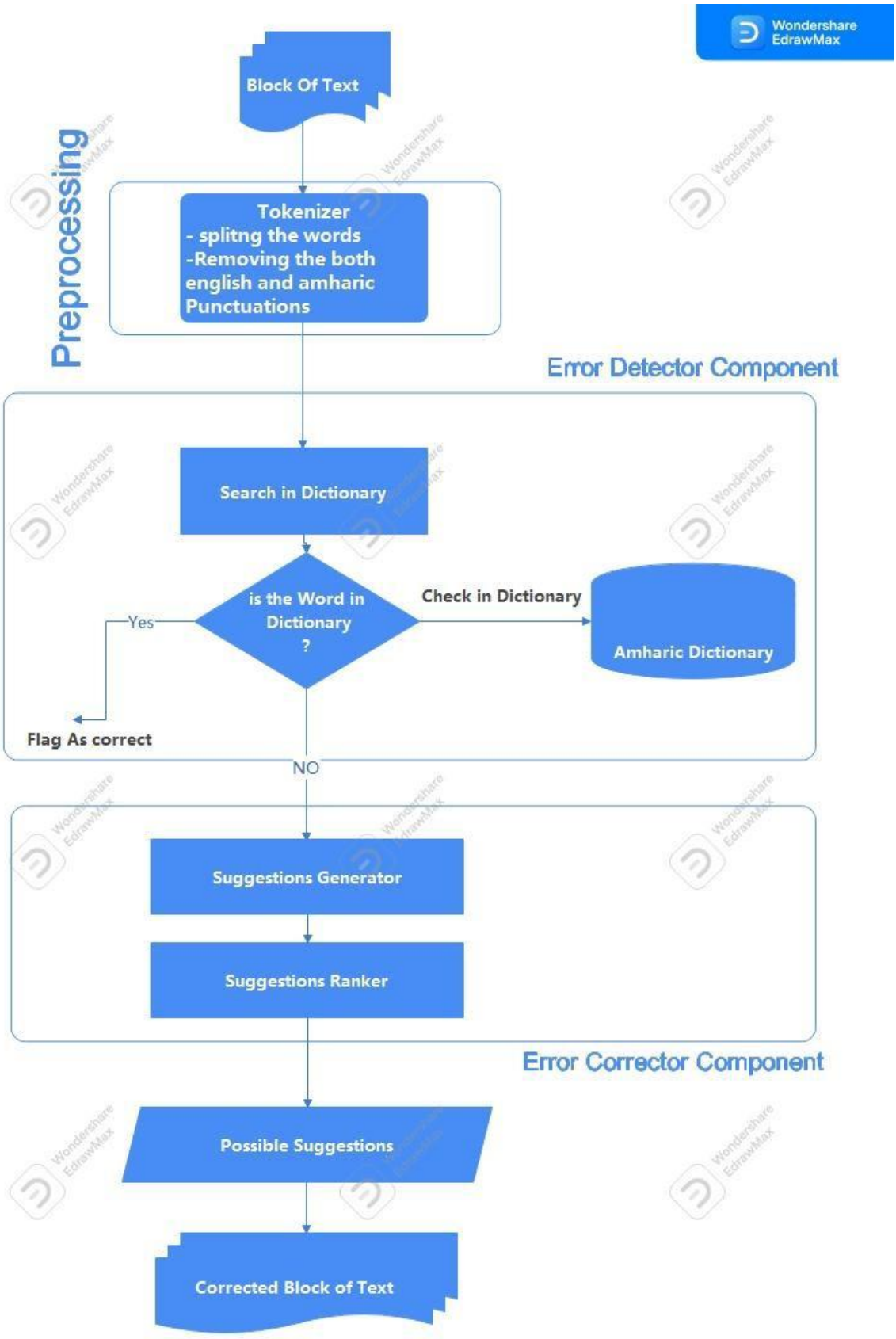

Figure 5 System Architecture of the model

4.2 Techniques used for Correction Spelling Error

4.2.1 Dictionary Lookup

A dictionary lookup is one of the popular algorithms which help to check if the given word is found in the dictionary or not. Dictionary is assumed to be the collection of correct words. And the main usage of the Amharic language dictionary that is used in this architecture is to store and retrieve the correct words. Besides a dictionary should allow the user to enter any new word which is believed a correct Amharic language word.

Dictionary lookup technique is a simple technique that checks the proposed word in the dictionary and decides if the word is correct or not. The non-standard word errors can be detected as mentioned above by checking each word from the dictionary. Every Dictionary has its characteristics like speed and storage. We can category dictionaries as big dictionaries and small dictionaries. The big dictionary can handle a large number of words but it takes time to search for a specific word. In addition, it consumes a large space to store the dictionary. In contrast, the small dictionary can only have a common word. And this can lead to incorrect word suggestions and even there can be no suggestion for the given misspelled word.

4.2.2 Sequence Matcher

The *difflib.get_close_matches ()* uses Sequence Matcher to return a list of the best "good enough" matches. Possibilities are a list of sequences against which to match a word (typically a list of strings). The best matches among the possibilities are returned in a list, sorted by similarity score, most similar first. *Difflib* is a very known library in python NLTK by its comparison and error detection and ranking capability. So, the researcher adapted this library to make the non-standard words Amharic spell checker and spell corrector.

A difflib.get_close_matches between two strings can be computed that gives a measurement of the number of steps needed to turn one string into the other (Doug Blank, 2012).

- difflib.get_close_matches (“ዩ ኒ ፍ”, “ዩ ኒ ሴፍ”) => 1 (deletion)
- difflib.get_close_matches (“ዩ ኤፍን ”, “ዩ ኤን ”) => 1 (insertion)
- difflib.get_close_matches (“ካ ፒታል”, “ካ ቢታል”) => 1 (substitution)

4.3 Test and Prototype Evaluation Procedure

4.3.1 Sampling Size

The researcher takes a sample size from the number of words that are stored in the Amharic Word Dictionary. And this is done using a formula defined below. (Slovin, 1960).

The sample size (n) is computed by this formula

$$n = \frac{\quad}{\quad}$$

Where: N = word size=342625

e = margin of error (1%-5%)

n = sample size=1100

The researcher collects 1100 sample words fromWalta television official website a report entitled with "ሀላ ት ሺሕ አ ሥዙ አ ራ ተ ኛ ውየ ል ደ ት በ ዓ ል ". The sample words were used to test the system developed by the researcher. To test the suitability of the system the researcher uses 20 users and 1100 words at the same time. A proper questionnaire is also prepared to get the feedback of the users when they use the system.

4.3.2 Data collection

The Amharic language word corpus is taken from Biniam, 2009. And this helps the researcher to design and model the non-standard Amharic language spell checker and corrector as the main input. The detail of the words is shown in Table 3. The researcher prepare a prototype designed using Tkinter which is the tool used to design (Generate) python User interface codes. According to Biniam, 2009 the corpus is prepared from different private and governmental newspapers, Books, and Different Documents.

The researcher tries to test the system designed as well as the corpus used to use some individuals who are Amharic language speakers. And those individuals have different professions and educational backgrounds (See Table 1).

The researcher tries to test the designed system using individuals who use Amharic language writings frequently as their primary job. The experiment is conducted using the prototype designed and installed in the Windows Operating system.

Table 1 Detail of the surveyor

No	Educational Background	Profession	Sex	
			Male	Female
1	Grade 10th completed	Private Sector	2	8
2	Diploma	Secretary	0	10
3	Degree graduated	Lawyers	5	0
		Total	25	

Out of the 25 individuals10 of them are grade 10 Completed with 2 Male and 8 Female, 10 of them TVET(Diploma) graduates 0 Male and 10 Female, 5 of them are Degree graduates with 5 Male and 0 Female. Each individual is requested to test 20 distinct words of his/her preferences. This makes the test data total 1100 words. The experiment is conducted on an individual's computer writing Amharic character on the proposed prototype.

4.4 Corpus Description

4.4.1 List of words of the Amharic corpus

The researcher uses the corpus which is developed by (Biniam, 2009). The researcher tried to figure out information about word length and what is the average word length of the Amharic language.

Table 2 List of Top 20 Distinct Words

No	Words
1	ፕሮፌሰ ር
2	ኤችአ ይቪ
3	እ ስ ፓን ያ
4	የ ኒ ቨ ር ሲቲ
5	ኤድስ
6	የ ስ ታት ቲክ ስ
7	ኮ ምፒዩ ተ ር
8	ሪ ፑብሊክ
9	በ ሻ ይረ ሱ
10	ኢን ተ ር ኔ ት
11	ጆር ጅ
12	ፋ ብሪ ካ
13	ፕሬ ስ
14	ፖሊሲ
15	ሶ በ ክ ሆተ ፕ
16	ቪዛ
17	ኢጣል ያ
18	ከ ኤች·አ ይ·ቪ
19	ጳ ጳ ሳ ት
20	ሄ ር ኩሌስ

4.4.2 Word Lengths of the Amharic corpus

The researcher tried to calculate the number of words of the Amharic corpus adapted from Amharic Biniam G. (2009). The researcher used Microsoft Excel to find the statistical information like average word length from the corpus which is developed by Amharic Biniam G. and the pie diagram shown in Figure 6 and Table 3. The researcher gets information about which word length of the Amharic language. The table below shows the length of the corpus.

Tabl 3 Word Lengths of the Amharic corpus

Word Length	Total
Total Words	342625
Letter 2	5208
Letter 3	28193
Letter 4	62502
Letter 5	82594
Letter 6	74433
Letter 7	50041
Letter 8	25260
Letter 9	9903
Letter 10	3179
Letter 11	893
Letter 12	256
Letter 13	90
Letter 14	34
Letter 15	22
Letter 16	6
Letter 17	3
Letter 18	4
Letter 19	3

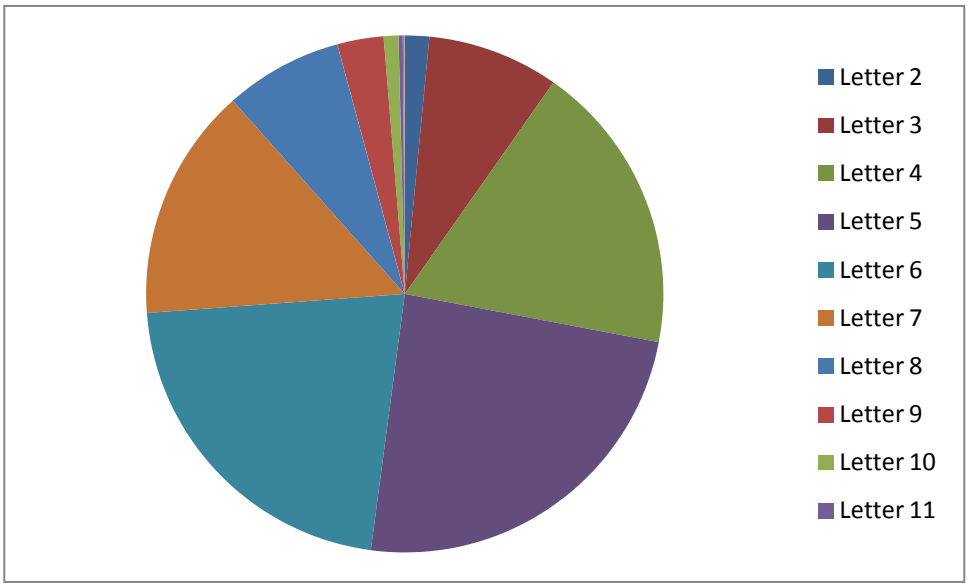

Figure 6 Word Lengths of the used corpus

The result of the tables, Table 2, Table 3 and the chart show Amharic Language speaker uses most of the time 5 letters words.

4.5 Presentation, Analysis, And Interpretation Of Data

4.5.1 User Interfaces

The user interface of the system is designed using PAGE. PAGE is a drag and drop GUI generator for python and Tkinter which generates python modules that display a relatively simple GUI constructed from Tk and tkk widget sets using the place, Geometry Manager. Since PAGE is a cross-platform tool running on any Operating System in which Tcl/Tk has been installed users that uses the system can run it in any operating system.

The user interface of the system is designed using different controls like Text Box, Label, Buttons, List Boxes, and Frames. This makes the System User interface easy to use and easy to understand. Besides the researcher tries to make the system in a local language that is the Amharic Language.

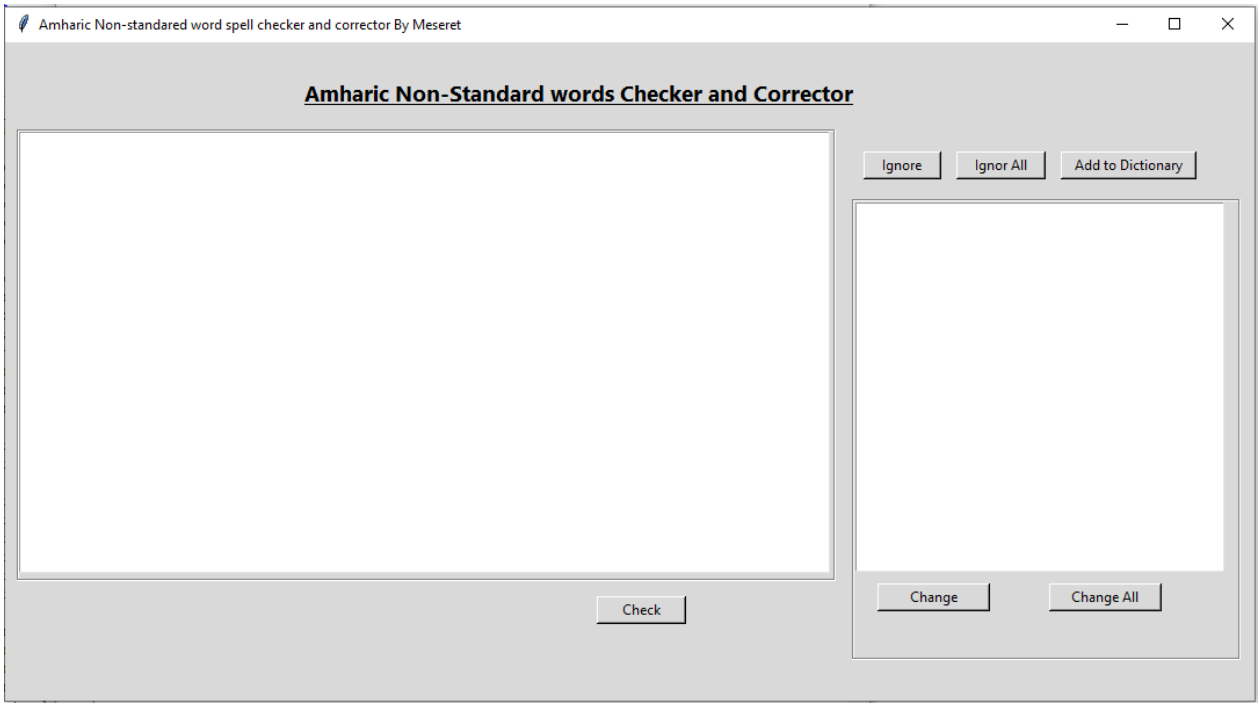

Figure 7 The System User Interface Design

4.5.2 Model Quality: Measuring Detection Accuracy

Measuring the model was metrics based on confusion matrix such as:-

Precision

Precision calculates the ability of a classifier to not label a true negative observation as positive.

$$Precision=TP/(TP+FP)$$

Recall (Sensitivity)

Recall calculates the ability of a classifier to find positive observations in the dataset.

$$Recall=TP/(TP+FN)$$

Accuracy

Accuracy (ACC) measures the fraction of correct predictions. It is defined as "the ratio of correct predictions to total predictions made".

ACC = (TP + TN) / (TP+TN+FP+FN)

Where TP- True Positive**, TN-** True Negative, **FP-** False Positive, **FN-** False Negative

Based on the above measuring mertrics the researcher use a 943 as a sample to test the model. The model able to detect 122 errors. Out of those errors the following categorization were there.

	True Positive	True Negative
Predicted Positive	821	50
Predicted Negative	20	122

Precision

Precision	0.9426	PPV = TP / (TP + FP)

Accuracy

Accuracy	0.9309	ACC = (TP + TN) / (P + N)

Recall

TP/(TP+FN) – 821(/821+122) = 0.870626

Other results such as

False Positive Rate	0.2907	FPR = FP / (FP + TN)
False Discovery Rate	0.0574	FDR = FP / (FP + TP)
False Negative Rate	0.0238	FNR = FN / (FN + TP)

As we can see from the above figures the model precision, accuracy and recall has 0.94, 0.93 and 0.87 respectively.

4.5.3 Quality of Design Results

The quality of design focused on ISO9241-11 usability metrics: ISO 9241 criteria are introduced to guide the quality of design in this study. As a result that, the results of how these criteria influence the design work are presented in this section.

4.5.4 Usability of the System

The researcher uses a questionnaire to be filled out by different respondents. The assigned values as strongly agree = 5, agree = 4, Neither Agree nor Disagree = 3, disagree = 2 and strongly disagree = 1. The table below shows the evaluation results.

Table 4 User's responses to statement I am satisfied with how easy it is to use this system.
How data is entered and output is suited to the tasks I want to perform with the Application.

Respondent	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
Grade 10 Completed	80 %	20 %	0 %	0 %	0 %
TVET (Diploma)	80 %	20 %	0 %	0 %	0 %
Degree Graduates	90 %	10%	0 %	0 %	0 %

Table 4 shows that respondents who have already completed the Degree Graduate program reported much higher satisfaction compared to TVET program and Grade 10 completed. These Grade 10 and TVET completed respondents have lesser satisfaction than the other respondents with how easy to use the system, particularly on how the data is entered.

Table 5 User's responses to statement in a given screen, I find all of the Information I need in that situation.

Respondent	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
Grade 10 Completed	90%	0%	10 %	0 %	0 %
Diploma (TVET)	90%	10 %	0 %	0 %	0 %
Degree Graduate	90 %	0%	10 %	0 %	0 %

According to Table 5 from the grade 10 respondents 90% replays Strongly Agree and 10% neither replays Neither Agree nor disagree. From the Diploma graduates, they respond 90% and 10% as Strongly Agree and Agree respectively. And from the Degree respondents, 90% replays Strongly Agree and 10% neither replays Neither Agree nor disagree.

And from the above figure, we can conclude the demo has almost everything on the screen that needs to check Amharic language spell language.

Table 6 User's responses to statement in a given screen, I find all of the Information I need in that situation.

Respondent	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
Grade 10 Completed	100%	0 %	0 %	0 %	0 %
Diploma (TVET)	100%	0 %	0 %	0 %	0 %
Degree Graduate	100%	0 %	0 %	0 %	0 %

As shown in the above Table 6, 100% of respondent replied that they Strongly agree on self-descriptiveness questioner**.** They understand immediately with the message displayed by the application when they are using the system.

The reason they understand immediately is because the application developed by the researcher is in local language Amharic and Some easy English words. The research can understand developing application in local languages helps to easily understand and can minimize committing errors.

Table 7 Users responses to statement the messages output by the Application always appear in the same screen location

Respondent	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
Grade 10 Completed	100%	0 %	0 %	0 %	0 %
Diploma (TVET)	100%	0 %	0 %	0 %	0 %
Degree Graduate	100%	0 %	0 %	0 %	0 %

As shown in Table 7 All the respondent replied that they are strongly agreed to statement the messages output by the application always appear in the same screen. In the developed prototype the suggestion words always displayed in one screen the is always on screen. And this helps the user to not confuse and to read the suggestion easily.

Therefore, it is possible to suggest that the application is message outputs always appear in the same screen location and users can find it everything messages and suggestion in the user's desire. Conformity is an important aspect of the user interface design because it helps the user to become familiar with using and learning the system. It also improves the efficiency and effectiveness of system usage, meanwhile reducing the percentage of error occurrence. The

researcher designed the system interface with unified visualized standards to provide the conformity of the system.

As a result of that all of the respondents strongly agree with the user expectations of the system.

Table 8 Users responses to statement the Application for Amharic non-standard word checker and correction reduces the input misspelled and effective for writing text

Respondent	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
Grade 10 Completed	80%	20 %	0 %	0 %	0 %
Diploma (TVET)	80%	20 %	0 %	0 %	0 %
Degree Graduate	100%	0 %	0 %	0 %	0 %

According to the above Table 8, 80% of the 10 Grade, TVET and 100% of Degree Graduated respondents replays that they strongly agree on the idea the system can reduce the input misspelled writing text, 20% of 10 Grade, TVET complete and Degree Graduate agree on the system correction and making free of misspell input.

Table 9 Users responses to statement I am able to effectively complete my Typing using this system with a short period of time

Respondent	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
Grade 10 Completed	80%	20 %	0 %	0 %	0 %
Diploma (TVET)	100%	0 %	0 %	0 %	0 %
Degree Graduate	100%	0 %	0 %	0 %	0 %

Regarding the Effectiveness of the system the data in the above Table 9, respondents result show that 80% of grade 10, 100% of Diploma and Degree Graduate Strongly agree that they are able to effectively complete their typing using the system with a short period of time. 20% of 10 Grade are in agreement with the system's effectiveness in typing using the application. From the above table, the researcher had understood that almost all of the respondents are strongly agreed on their effective completion of typing using the application. From this it is possible to suggest that the system is manageable and suitable for learners' Therefore the above table answers the research question.

Table 10 Users responses to statement I find it easy to use the commands

Respondent	**Strongly Agree**	**Agree**	**Neither Agree nor Disagree**	**Disagree**	**Strongly Disagree**
Grade 10 Completed	90%	0 %	10 %	0 %	0 %
Diploma (TVET)	90%	0 %	10 %	0 %	0 %
Degree Graduate	100%	0 %	0 %	0 %	0 %

According to the above Table 10, 90 % 10 Grade, 90% TVET, 100% Degree Graduate of the respondent indicates that strongly agree on the suitability of the command easiness for learners. 10% of 10 Grade, 10% Diploma, 0% Degree Graduated also neither agree nor disagree on the application command they found easy for users.

Generally, the system spelling checker model is effective, efficient and reduce the time for writing Amharic words in generating relevant suggestion in spell checker and correction as a result of this the researcher scores 92 accuracy from a sample of 1100 words of the population of words suggested by the application spelling checker and correction.

From this finding, one can understand that the accuracy of the systems modeling for Amharic spelling checker and correction uses python in desktop application scores satisfactory result in generating relevant word.

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORKS

5.1 Conclusion

The main goal of the researcher and the study is to design a model, implement and develop a prototype for non-standard word Amharic language error spelling checker and corrector. The researcher come up with a full-fledged new architecture that can answer the question stated in chapter one. To answer the research questions in the first chapter the researcher used different methods and tools. As a result the model precision, accuracy and recall has 0.94, 0.93 and 0.87 respectively. A prototype is also developed to show if the architecture works. And the researcher tried to measure the usability of the prototype using the ISO 9241 usability engineering standards and participating users from different professions and educational backgrounds and it achieved the result to be 92%. Still, now no word processing software supports non-standard error Amharic language spell checker. Due to this the Amharic language users are using the words in their sense, not based on the standards. To mention one example the word "ማክ ክ " is considered the correct word according to the research conducted by Biniam, 2009. But some groups use the word "ማክስ " and other groups can write the word according to their common sense. Nowadays according to the researcher survey secretaries write 20-30 pages on average per day. During this writing time, the secretaries commit an error and no software exists to detect the error and gives probable suggestions. The use of a spellchecker for different tasks in Desktop and laptop computers is one of the many applications undertaken by various people for a different purposes. Developing a standalone application can minimize the amount of time and energy losses to find and correct the misspelled word. Process of finding the relevant word that a user intends to write after the misspelled word based on a lexicon and statistical information obtained from the corpus.

To accomplish the task of the Spelling checker process for non-standard error Amharic language, a good collection of words is necessary. However, since there is such a prepared corpus for the Amharic language used from Amharic Wikipedia, the researcher used it as a dictionary for the prepared prototype. In addition to this, the researcher permits the user to add words to the dictionary which considered as correct words but not added to the dictionary. The Analyses have

been done on the developed application. In this work, the researcher concluded that the implementation, testing and evaluation of systems modeling for non-standard word Amharic spelling checker and correction has fulfilled the objectives of providing a tool with a reasonably good suggestion support in Amharic language for spelling checker text entry.

5.2 Future Works and Recommendation

In this study, the researcher conducted the first ever work in system modeling for Amharic spelling checker and correction in standalone desktop application with python. However, the researcher is aware but failed to add more new Amharic words and make the prototype user interface to add more features. Because of python GUI (Tkinter) is new for the researcher we cannot add more controls that can change the correct word immediately.

REFERENCE

[1] Mulu. Alemebante, Vishal. Goyal, "Amharic Text Predict System for Mobile Phone," *Int. J. Comput. Sci. Trends Technol* 3,4, 2015

[2] Cambria. Erik, Bebo White, "Jumping NLP curves: A review of natural languageprocessing research," *IEEE Computational intelligence magazine* 9,2, 2014.

[3] Cobley. R ,R. Parry, "Spell check," Nursing times 93,16, 1997

[4] Bassil.Youssef,"Parallel spell-checking algorithm based on yahoo! n-grams dataset," arXiv preprint arXiv:1204,0184, 2014.

[5] Sidorov. A. A,"Analysis of word similarity in spelling correction systems," Program and Computer Software 5, no.4 , 1979

[6] Blank. Douglas S, "Research Areas," Education 610, 1997

[7] Wasala.Asanka, Ruvan .Weerasinghe, Randil. Pushpananda, Chamila .Liyanage, and Eranga. Jayalatharachchi, "A data-driven approach to checking and correcting spelling errors in Sinhala," Int. J. Adv. ICT Emerg, Reg 3, no. 01 , 2010

[8] Rashmi. M, C. D. Manning, P. Raghavan, and H. Schütze, "Introduction to information retrieval systems," Int. J. Recent Innov. Trends Comput. Commun 3, no. 4 , 2015

[9] Dhanabalan. T, Ranjani. Parthasarathi, T. V. Geetha, "Tamil spell checker," In Sixth Tamil Internet 2003 Conference, Chennai, Tamilnadu, India, 2003.

[10] Osman. Omer, Yoshiki. Mikami, "Stemming Tigrinya words for information retrieval," In Proceedings of COLING 2012: Demonstration Papers, pp. 345-352, 2012.

[11] Bard. Gregory V, "Spelling-error tolerant, order-independent pass-phrases via the Damerau-Levenshtein string-edit distance metric," Cryptology ePrint Archive, 2006

[12] Damerau. Fred J, "A technique for computer detection and correction of spelling errors," Communications of the ACM 7, no. 3, 1964

[13] Naseem. Tahira, "A hybrid approach for Urdu spell checking," Master of Science (Computer Science) thesis at the National University of Computer & Emerging Sciences (2004).

[14] Kukich. Karen,"Techniques for automatically correcting words in text," Acm Computing Surveys (CSUR) 24, no. 4, 1992

[15] Fossati. Davide, Barbara Di Eugenio, "A mixed trigrams approach for context sensitive spell checking," In International conference on intelligent text processing and computational linguistics, pp. 623-633. Springer, Berlin, Heidelberg, 2007.

[16] Peterson, James L. "Computer programs for detecting and correcting spelling errors." Communications of the ACM 23, no. 12, 1980

[17] Tarniceriu. Adrian, Bixio. Rimoldi, and Pierre. Dillenbourg, "HMM-based error correction mechanism for five-key chording keyboards," In 2015 International Symposium on Signals, Circuits and Systems (ISSCS), pp. 1-4. IEEE, 2015.

[18] Lauriola. Ivano, Alberto. Lavelli, Fabio. Aiolli, "An introduction to deep learning in natural language processing: models, techniques, and tools," Neurocomputing 470, 2022

[19] Pal. U, Pulak K. Kundu, , Bidyut Baran Chaudhuri, "OCR error correction of aninflectional indian language using morphological parsing," J. Inf. Sci. Eng. 16, no. 6, 2000

[20] Liang, Hsuan Lorraine, "Spell checkers and correctors: A unified treatment," PhD diss., University of Pretoria, 2009

[21] Randhawa. Er, Sumreet Kaur, Er Charanjiv Singh Saroa,"Study of spell checking techniques and available spell checkers in regional languages: a survey," International Journal For Technological Research In Engineering 2, no. 3, 2014

[22] Faulk. Ramon D, "An inductive approach to language translation," Communications of the ACM 7, no. 11, 1964

[23] Neha. Gupta, and Pratistha Mathur, "Spell checking techniques in NLP: a survey." International Journal of Advanced Research in Computer Science and Software Engineering 2, no. 12 (2012).

[24] Khelifi. Abran A, "A, Suryn," W," Usability Meanings and Interpretations in ISO Standards" Software Quality Journal 11, 2003

[25] Ganfure. Gaddisa Olani. Dida Midekso, "Design And Implementation Of Morphology Based Spell Checker," vol 3, 2014

[26] Kaur. Amanjot, Paramjeet Singh, Shaveta Rani, "Spell Checking and Error Correcting System for text paragraphs written in Punjabi Language using Hybrid approach." International Journal of Engineering and Computer Science 3, no. 09 , 2014

[27] Attia.Mohammed.Pavel Pecina, Younes Samih, Khaled Shaalan, Josef Van Genabith,"Improved spelling error detection and correction for Arabic," In Proceedings of COLING 2012: Posters, pp. 103-112. 2012.

[28] Gupta. Neha, Pratistha Mathur, "Spell checking techniques in NLP: a survey," International Journal of Advanced Research in Computer Science and Software Engineering 2, no. 12, 2012

[29] Kothari. Chakravanti Rajagopalachari,Research methodology: Methods and techniques. New Age International, 2004.

[30] Hevner. Alan R, Salvatore T. March, Jinsoo Park, Sudha Ram, "Design science in information systems research," MIS quarterly, 2004

[31] Hevner. Alan, Samir Chatterjee, "Design science research in information systems," In Design research in information systems, pp. 9-22. Springer, Boston, MA, 2010

[32] Peffers. Ken, Tuure Tuunanen, Marcus A. Rothenberger, Samir Chatterjee. "A design science research methodology for information systems research," Journal of management information systems 24, no. 3, 2007

[33] Wieringa, Roel J. Design science methodology for information systems and software engineering. Springer, 2014.