APPLICATION OF HYBRID APPROACH FOR WOLAITA LANGUAGE PART OF SPEECH TAGGING

MSc. THESIS

BIRHANESH FIKRE SHIRKO

HAWASSA UNIVERSITY- INSTITUTE OF TECHNOLOGY

FACULTY OF INFORMATICS

HAWASSA, ETHIOPIA

August, 2020

APPLICATION OF HYBRID APPROACH FOR WOLAITA LANGUAGE PART OF SPEECH TAGGING

BIRHANESH FIKRE SHIRKO

ADVISOR: WONDWOSSEN MULUGETA (PHD)

A THESIS SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCES,

INSTITUTE OF TECHNOLOGY, SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY

HAWASSA, ETHIOPIA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN COMPUTER SCIENCES

(SPECIALIZATION: COMPUTER SCIENCE)

HAWASSA, ETHIOPIA

AUGUST, 2020

ADVISORS' APPROVAL SHEET

SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY

This is to certify that the thesis entitled — "**Application of Hybrid Approach for Wolaita Language Part of Speech Tagging"** submitted in partial fulfillment of the requirements for the degree of **Master's** with specialization in **Computer Science,** the Graduate Program of the **Department of Computer Science,** has been carried out by Birhanesh Fikre Id. No PGCSR/005/10, under my supervision. Therefore I recommend that the student has fulfilled the requirements and hence hereby can submit the thesis to the department.

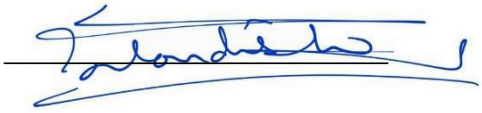
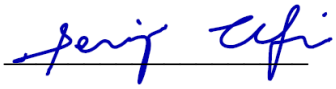
Wondwossen Mulugeta (PHD)		August 16, 2020
Name of advisor	Signature	Date

EXAMINER'S APPROVAL SHEET

SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY

As members of the Examining Board of the Final MSc Open Defense, we certify that we have read and evaluated the thesis prepared by Birhanesh Fikre Shirko entitled **— "Application of Hybrid Approach for Wolaita Language Part of Speech Tagging"** and recommend that it be accepted as fulfilling the thesis requirement for the degree of Masters of science in computer science.

Name of Chairman	Signature	Date
Wondwossen Mulugeta (PHD)		August 16, 2020
Name of Major Advisor	Signature	Date
Name of Internal Examiner	Signature	Date
Dr. Dereje Teferi		August 17, 2020
Name of External Examiner	Signature	Date

Final approval and acceptance of the thesis is contingent upon the submission of the final copy of the thesis to the Department of Graduate Council (DGC) of the candidate's major department.

Stamp of SGS

Date: _____

DEDICATION

This thesis is dedicated to my father Fikre Shirko and my mother Zenebech Dea; they are my everything throughout my life.

DECLARATION

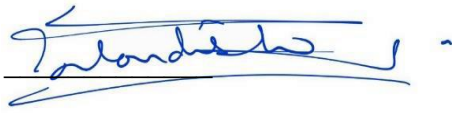
I hereby declare that this MSc thesis is my original work and has not been presented for a degree in any other university, and all sources of material used for this thesis have been properly acknowledged.

Name: Birhanesh Fikre

Signature: _____

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

Name: Wondwossen Mulugeta (PhD)

Signature: 

Place and Date of Submission: Hawassa, June, 2020

ACKNOWLEDGEMENT

Overhead all, my truthful thank and praise goes to the Almighty GOD for achieving this study work Successful. I would like to express my heartfelt thankfulness to my advisor, Wondwossen Mulugeta (PhD) for his close supervision, and constructive suggestion during my research work. He has been devoting his time and providing all necessary relevant information to carry out the research. I will extend my thanks to the Wolaita Sodo University for sponsoring this study chance. In addition, my appreciations also go to all staffs of Hawassa university and Faculty of informatics specially department of computer science. I am greatly indebted to my instructors Dr.Tesfaye Bayu, Dr. Wondwossen Mulugeta and Dr. Dereje Teferi, Dr.Basant T., Dr. Anitha V. for their support during my thesis work. I am grateful thanks to language experts Mr.Temsigen Ajaja and Mr.Kuleno Kolicha. Special thanks go to my parents, brothers, and sisters for their unreserved all rounded support, endless prayers, and love they have for me.

Table of Contents

List of Table

Table **Page**

List of Figure

Figures **Page**

LIST OF ABBREVIATIONS AND ACRONYMS

AI----------Artificial Intelligence

HMM----------Hidden Markov Model

NLP----------Natural Language Processing

NLTK----------Natural Language Toolkit

PoS----------Part-of-Speech

PoST----------Part-of-Speech Tagging

SOV----------Subject Object Verb

TBL----------Transformation Based Learning

TDS----------Training Data Set

TS----------Test Data set

WL----------Wolaita Language

ABSTRACT

The aim of this research is to develop part-of-speech tagger for Wolaita Language using hybrid approach. Part of speech tagger is one of the subtasks in natural language processing (NLP) applications which is vital for other NLP tasks, like parser, machine translator, speech recognizer and search engines. It is a process of labeling a corresponding part of speech (PoS) tag for a word that defines how the word is used in a sentence. The PoS tagging for Wolaita language is not sufficient yet to be used as one important component in other natural language processing (NLP) applications. In this thesis, the development of part of speech tagger using hybrid approach that combines HMM and transformation based learning approaches is conducted for Wolaita language. In general, HMM model needs large data to increase the performance and the transformation based learning model learn rule based on the language features. The HMM model tags the words based on the optimal path for a given sequence of words and transformation based learning is a rule based model that tag the words based on rules; it learns rule directly from the training corpus without expert knowledge. The developed hybrid approach of Wolaita language PoS tagger uses HMM tagger as initial annotators and the rule based tagger as a corrector based on fixed threshold values. For implementation and experiment, the author used python programming and NLTK. For training and testing the model, 14,358 untagged Wolaita language words are collected from three different categories (Bible, Social media in Wolaita language (Wogetta FM 96.6) and Wolaita language department). The annotation of corpus performed manually by two language experts. For tagging purpose 26 PoS tag are identified based on the work of Berhanu H., work of wakasa (2008) and with help of language experts. From the entire corpus, 90% is used for training and the remaining 10% is used for testing purpose. The performance of the three taggers is tested by using different experiments. After experiment the researcher found that the performance of HMM, rule based and hybrid taggers shows 88.14%, 92.96% and 94.82% respectively. Generally, hybrid approach showed the better performance to assigning part of speech tag for Wolaita language sentences.

Key words: NLP, HMM, TBL, NLTK and Hybrid

CHAPTER ONE

1. INTRODUCTION

Natural language processing (NLP) is the ability of computer program to understand human language as it is spoken or written. NLP is the component of artificial intelligence (AI), which is automatic manipulation of natural language, like speech and text by computer software. Natural language can be applied to any language that human beings use to communicate with each other [1]. NLP has many application areas. Some of these are text-to-speech and speech recognition, natural language dialogue interfaces to databases, information retrieval, information extraction, document classification, document image analysis, automatic summarization, spelling and grammar checking, machine translation, Part of Speech (PoS) tagging, plagiarism detection, stemming and others[2].

In this research the focus is on part-of-speech tagging for Wolaita language. Advanced natural language processing application requires part of speech tagging as preprocessing since identifying the part of speech or word class of a token is very important to determine its morphology, pronunciation and even semantics. Part of speech tagging is the ground level step of the advanced NLP application like text to speech, information retrieval, information extraction, parsing, machine translation and other NLP application. Therefore, part of speech tagging is very important for NLP application, especially for languages that are under resourced. Similarly, for Wolaita language, to develop other high level NLP applications, part of speech tagging is one of the fundamental NLP tasks that need to be accomplished.

Wolaita language is one of the Northern Omotic languages that is spoken in the Wolaita Zone and some other parts of the Southern Nations, Nationalities, and People's Region of Ethiopia. It is also spoken in different cities of the country by the people from Wolaita and neighborhood Zones. The language has around 3.3 million native and dialective speakers [3]. However, computer applications that help to use the language in a more advanced way like text summarization, information retrieval and extraction, parsing and machine translation[2] are not available for this language. Works in natural language processing also showed that

these high-level applications require low level language processing systems like part of speech tagging. The aim of this study, therefore, is to put an effort in preparing the groundwork in advance as part of speech tagging that enable to the next phase of natural language processing application developments. In Ethiopia, there are more than 80 different languages and among these languages, only four languages currently working with technological center especially in the telecom communication system. So, this study tries to contribute to the advancement to overcome the problems mentioned above. To use computers for understanding and manipulation of Wolaita language, few works are conducted in this language. These include text-to-speech system for Wolaita language [4], speaker dependent speech recognition for Wolaita Language[5], development of a stemming algorithm for Wolaita language[6] and development of longest-match based stemmer for texts of Wolaita language[3]. There are also other related researches that were conducted on other local language. Specially on Amharic language, some researches were conducted on PoS tagging by[7] and[8], but in the Wolaita language there is a beginning work PoS tagging is developed using small corpus in [9]. Hence, it is the aim of this study to develop automatic PoS tagger for Wolaita language to establish the base for future researchers who will have interest in the area of machine translation, information retrieval, text summarization as well as other NLP research for Wolaita language.

1.1. Statement of the Problem

Wolaita language is a language that has a huge number of speakers in Wolaita and neighboring people who are located in southern part of Ethiopia. According to Motomichi [10], all the west coast region of the Lake Abaya is considered to be an area where Wolaita is spoken. Some of these areas are: Gamo, Gofa, Dawuro, Kucha and Kullo. Wolaita language serves as a medium of instruction in primary schools and offered as one subject for high schools students in Wolaita zone. In addition, the Wolaita language department at Wolaita Sodo University offers bachelor degree in Wolaita language studies. PoS is very important to understand Wolaita language and construct the correct sentence structure according to the grammar basically for students and teachers. There is no automated spell-checker and grammar checker for Wolaita language; as the result, it is difficult to correct spelling error in the given sentence without the basic knowledge of PoS. Therefore, it is used to write correct

spelling when the users use the language for their academic activities like assignment, research and others. In this study, the researcher design PoST for Wolaita language, this study will be an input for advanced NLP research works.

There are some computational attempts on Wolaita language for different types of NLP application which does not cover all aspects of the language [3]. Lack of some of the basic processing tools might create a gap in processing the linguistic resources of Wolaita language with computers. This, in turn, creates challenge to understand and use the language for communication and transformation of knowledge in the digitized manner. There is Wolaita language corpus but it is not preprocessed or tagged corpus that can be used for PoS tagger and used by researchers in order to work on advanced NLP research on Wolaita language like machine translation, speech synthesis, question and answering, parsing, spell checker [2].

There are few computational linguistics work conducted in this language. Some of these are text-to-speech system for Wolaita language [4], speaker dependent speech recognition for Wolaita Language [5], development stemming algorithm for Wolaita language[6], development of longest-match based stemmer for texts of Wolaita language[3] and POS tagging research developed for Wolaita language by using small corpus[9]. Most of the time, Hidden Markov Model (HMM) approach tends to give better performance with large annotated corpus. But Birhanu.H [9] used small corpus for training the tagger, used two approaches separately and small number of tagset in his attempt to develop PoS tagger for Wolaita language. The researcher developed PoST for Wolaita language using HMM and CRF method in supervised and unsupervised machine learning approach. For training and testing purpose he used 200 sentences which are manually tagged. The research result showed that the accuracy of HMM and CRF is 83.58% and 74.63% respectively and HMM tagger perform better than CRF tagger.

In[11] the researcher designed and developed part of speech tagger for Kafi-noonoo language using hybrid approach of HMM and rule based tagger. Kafi-noonoo language is one of an Omotic family and its uses Latin script which is morphological complex like its neighbor Wolaita language[12]. The author identified 34 tagset and 354 sentences are used for training and testing taggers. Hence, he concluded that hybrid tagger at sentence level outperforms

HMM and rule-based tagger taken individually. The researcher[13] combined two approaches that are rule based and stochastic approach to model the PoS tagger for Afaan Oromo language and the hybrid model improved the accuracy to 98.3%. Thus, it is found that the hybrid approach outperforms the individual approaches. Likewise, Meryeme Hadni et al,[14] used a hybrid approach that combine rule based and HMM tagger , they got the better result(98%) in the hybrid approach for non-vocalized Arabic text.

Based on the above mentioned research attempts, this research aims to combine two approaches to develop a hybrid part of speech tagger for Wolaita language rather than using individual approaches separately plus relatively large corpus size is used in order to get better accuracy.

1.2. Research questions

1. Which linguistic feature of Wolaita language would be challenge for POS tagger of Wolaita language?
2. To what extent the Wolaita language PoS tagger works using hybrid approach?

1.3. Objectives of the Study

1.3.1. General Objective

The general objective of the study is to explore the application of a hybrid approach for the development of part-of-speech tagger model for Wolaita language.

1.3.2. Specific Objective

This research work has the following specific objectives to achieve the above mentioned general objective.

- ✓ To review literatures and analyze related documents in Wolaita language word categories, morphological properties and sentences structure to derive the PoS tags and the tag set for the language.
- ✓ To develop a Wolaita language corpus.
- ✓ To design an algorithm to develop a PoS tagger.
- ✓ To test and analyze the performance of the PoS tagger

- ✓ To draw conclusions from experimental results
- ✓ To provide recommendation for further research.

1.4. Scope of the study

The scope of the research is limited to studying the possibility of combining Hidden Markov Model and transformation based learning approach to design automatic PoS tagger for Wolaita language. The study is also limited to implement with HMM models of Viterbi algorithm and rule based model of transformation based learning. The study emphases on tagging of part of speech of the language.

1.5. Significance of study

This study can be used as a reference for researchers who are interested to develop on high level application of NLP for Wolaita language like text summarization, information extraction, machine translation, spell checking etc. Another significance of the study belongs to the people who want to learn Wolaita language as second language; it may help them to learn the word categories and grammar construction.

1.6. Organization of the Thesis

This thesis consists of six chapters. The first chapter describes the general introduction of the study including introduction of the study, statement of the problem, objectives, scope and significance of the research. Chapter 2 presents literature review and related works which are done by using hybrid approaches on part of speech tagger. Chapter 3 emphases on the study and assessment of the nature and structure of Wolaita sentences, word classes and tagset preparation. Chapter 4 presents the design of PoS tagger for Wolaita language. Chapter 5 presents experiment result. Lastly, in Chapter 6, conclusion and recommendation are presented.

CHAPTER TWO

2. LITERATURE REVIEW

2.1. INTRODUCTION

Part of speech tagging is one of the crucial applications in natural language processing (NLP). Natural language processing (NLP) can be defined as the automatic (or semi-automatic) processing of human language[15]. Natural Language Processing (NLP) is a tract of Artificial Intelligence and Linguistics, devoted to make computers understand the statements or words written in human languages. Natural language processing came into existence to ease the user's work and to satisfy the wish to communicate with the computer in natural language. Natural Language Processing basically can be classified into two parts i.e. Natural Language Understanding and Natural Language Generation which evolves the task to understand and generate the text[16].

2.2. NLP approaches

Natural languages processing applications can be experimented and developed using various approaches. Some of the approaches are knowledge based that seek human knowledge for the development of the rules and exceptions for language features handling. There are also other applications which try to learn language processing rules from sample data or training corpus. Some of these NLP development approaches are:

i. **Rule based approaches:** Develop the rules to process natural language based on known fact and exception. It uses linguistic grammar-based rules containing morphological, syntactical and lexical information to find tags. Rules to be used in the tagging process are either manually hand-crafted by linguistic professionals or automatically generated using machine learning algorithms[17].

ii. **Machine learning approach:** Machine learning is a branch of *artificial intelligence* that aims at enabling machines to perform their jobs skillfully by using intelligent software. The statistical learning methods constitute the backbone of intelligent software that is used to develop machine intelligence. Because machine learning algorithms require data

to learn, the discipline must have connection with the discipline of database[18]. Learn rules from examples and apply on new instance. Machine learning is a technology design to build intelligent systems. These systems also have the capability to learn from experience. It delivers results according to its experience. Machine learning techniques can be divided into various categories according to their purpose and the main categories are the following[19]: supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. The following figure present different machine learning techniques and their required data[18].

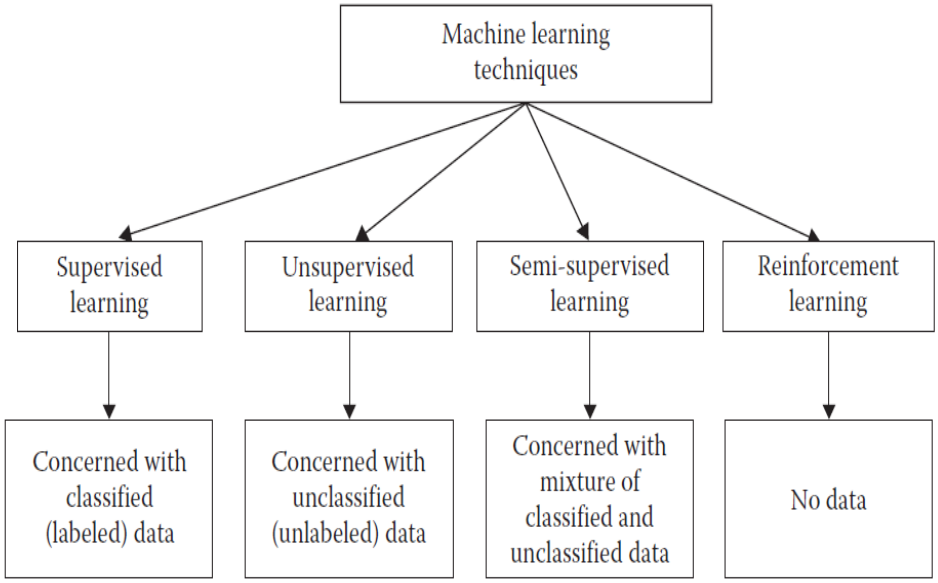


Figure 1 (***Machine Learning: Algorithms and Applications***): Machine Learning Techniques and their Required Data

a) Supervised approach-> learn by comparing with expected output. The main aim in supervised learning is to learn a model from classified training data that allows us to make predictions about unseen data. In supervised learning, the target is to infer a function or mapping from training data that is labeled or classified. The training data consist of input vector X and output vector Y of labels or tags. A label or tag from vector Y is the explanation of its respective input example from input vector X[18].
b) Semi-supervised approach-> is a machine learning approach that combine labeled and unlabeled data for training - typically a small amount of labeled data with a large amount of

unlabeled data. This method is so called "semi-supervised approach" because it uses data required for both supervised and unsupervised approaches[20].

c) Unsupervised approach-> blind learning. Create knowledge by association rather than predefined output. The unsupervised learning algorithms learn few features from the data. When new data is introduced, it uses the previously learned features to recognize the class of the data. It is mainly used for clustering and feature reduction[19].

d) Reinforcement learning→ the reinforcement learning method aims at using observations gathered from the interaction with the environment to take actions that would maximize the reward or minimize the risk. In order to produce intelligent programs (also called *agents*), reinforcement learning goes through the following steps[18]:

- ✓ Input state is observed by the agent.
- ✓ Decision making function is used to make the agent perform an action.
- ✓ After the action is performed, the agent receives reward or reinforcement from the environment.
- ✓ The state-action pair information about the reward is stored.

2.3. Part of Speech Tagging Method

Each language has its parts of speech for instance verb, noun, adjective…etc. Part of speech is a basic category of words allowing their function in sentence. It is also called lexical categories, grammatical categories or word categories. Tag is a string used as a label in part of speech tagging. Tagset is the collection of tags from which the tagger is supposed to select to attach to the relevant word. The set of tags used for a specific task is known as a tagset. Parts-of-Speech (PoS) tagging is a method for annotation of lexical categories to every word in input sentence.

PoS tagging is extensively used for linguistic analysis of input text. It is very important task and preprocessing step for all the natural language processing activities. A PoS tagger takes a sentence from input data and assigns a unique parts of speech tag to each lexical item of the sentence. Parts-of-Speech (PoS) tagger has been developed for many languages as a part of Natural Language Processing[21]. The key concern of developing part of speech tagger for any language is to improve accuracy of tagging and remove ambiguity in sentences due to language structure. PoS tagger is relevant to improve accuracy if the researcher develops advanced NLP application like parsing, speech recognition and machine translation. It is important to learn

grammar construction and understand the language. This thesis focuses on developing PoS tagger for Wolaita language. There are different approaches for PoS tagging. The following figure demonstrates different POS tagging models[22].

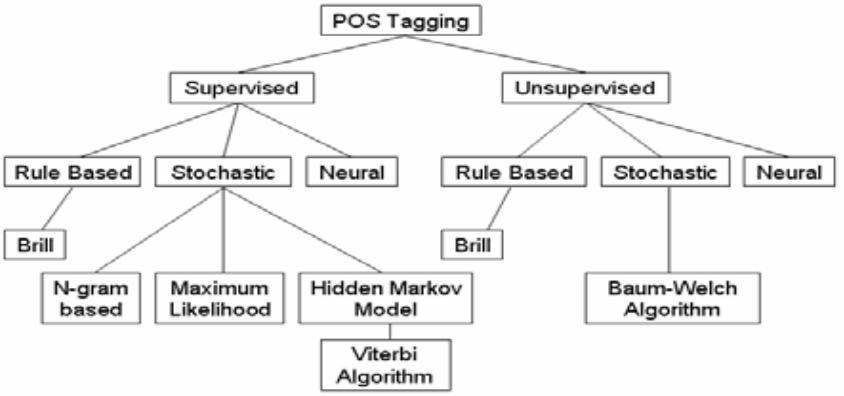


Figure 2(Comparison of different PoS Tagging Techniques for some South Asian Languages): Classification of PoS Tagging Model

From the above PoS tagging model we have planned to use the rule based (Transformation Based Learning (TBL)) and HMM by using Viterbi algorithm. Part of speech tagger model is a model that is used for development of tagger for a language. The Part of Speech (PoS) tagging refers to the process of assigning appropriate lexical category to individual word in a sentence of a natural language[23]. In overall, the part of speech tagger for the natural languages are developed using rule-based, probabilistic models and combination of both. These models are grouped into both supervised and unsupervised approaches. The following subsections describe PoS tagging model.

i) **Supervised Learning**: In supervised learning, already prepared corpus is used and according to that corpus output will be generated. It is easy to tag words in the sentence with supervised tagging.

ii) **Unsupervised Learning**: In unsupervised learning, first task is to train corpus and then tag the input sentence.

These methods are also sub-categorized into three types: Rule based Technique, statistical technique and transformation based technique.

1) **Rule Based Technique**: Rule-based taggers generally depend on rules hand-written by humans. Knowledge based taggers and usually rules built manually. These rules can be manually prepared rules by linguistic professionals and machine learned rules. The advantage of rule-based approach is, it is not dependent on the amount of the data in hand which enables it to work well on small corpus. It also gives better performance when applied on part of speech taggers developed for limited domains. The disadvantages of rule-based approach is inflexibility of adding and removing a rule from the set of rules defined, time taking, tedious and linguistic professional requirement.

2) Stochastic technique

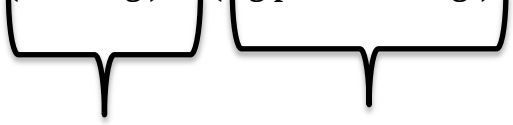
Stochastic method which can also be called statistical method is the method which is based on the statistical information or on the probability of occurrence of words. Any approach that uses probability or statistic information can be grouped under this approach[24]. Mostly it is used as corpus based technique. This method is built based on probability of word occurrence and the sequence of part of speech of previous instances.

Here, it is possible that two words have same spelling but possess different part of speech according to the context of sentence in which the words appear. So decision on the part of speech of a word depends on the probability distribution of the word with respect to the previous words and the statistical distribution generated from the training corpus [25]. Mainly there are three sub-techniques in statistical approaches: Hidden Markov Model, Maximum Entropy Markov Model and Conditional Random Fields.

a) Hidden Markov Model

The idea behind Hidden Markov Model tagger is that "pick the most likely tag for the word" approach[26]. Hidden Markov Model (HHM) is probabilistic. It calculates the most likely tag of a word based on the context of the word and its immediate neighbors. It is the statistical model which is mostly used in PoS tagging. The general idea is that, if there are sequences of words, each with one or more potential tags, and then the most likely sequence of tags is determined by calculating the probability of all possible sequences of tags, and then choosing the sequence with the highest probability. It calculates the probability based on Bayes rule.

P(word/tag) * P(tag/previous n tags)…….2.1



Most frequent tag N-gram(a prior)

(Likelihood)

At the training phase of HMM based PoS tagging, observation probability matrix and tag transition probability matrix are generated[27].

According to [28] the parameters needed to define HMM are:

- ✓ States: a set of state
- ✓ Observation sequence: a set of legal accepting states
- ✓ Transition probability: probability distribution that govern the transition from one state to another state.
- ✓ Observation likelihoods: a set of observation likelihoods
- ✓ Initial distribution: an initial distribution of states.

b) Maximum Entropy Markov Model

Maximum Entropy Markov Model is a conditional probabilistic sequence model. It can represent multiple features of a word and can also handle long term dependency. It is based on the principle of maximum entropy which states that the least biased model which considers all known facts is the one which maximizes entropy[29].

Maximum likelihood method is the simplest one in stochastic approach which assigns the most frequent parts-of-speech tag found in the training data to a text in a non-tagged data. The probability of a tag to be assigned for a word can be calculated by counting every word with a specific tag and dividing it with the number of occurrences for this particular tag[11]. This gives the conditional probability of the word given the tag. The major problem of maximum likelihood method is that it does not consider the context of a word in a sentence when assigning the most appropriate tag [1].

c) Conditional Random Fields

Conditional random field is a probabilistic framework for labeling and segmenting data. It is a form of undirected graphical model that describes a single log-linear distribution over label classifications given a particular observation sequence. CRFs define conditional probability distributions P(Y|X) of label sequences given input sequences. The probability of a particular label sequence Y given observation sequence X to be a normalized product of potential functions each of the form[30].

3) Transformation Based Technique

Transformation-based learning is an error-driven approach to induce the retagging rules from a training corpus. The learning algorithm starts by building a lexicon containing each word with all possible tags and the frequency of each tag. Then, it maintains two versions of the corpus; gold standard corpus that contains word/tag and training corpus that contains only words. Next, it assigns the most frequent tag to each word in the training corpus which referred as initial state tagging. After that, it compares the initially annotated corpus with the gold-standard corpus and the largest error class. Focusing on that error class, it applies a set of predefined templates to correct it and chooses the rule with highest gain; gain means number of corrections in the training corpus. That rule is stored in a list after being applied to the training corpus. Then, it calculates the largest error class from the updated training corpus again and so on until no correction could be made.[31].

From the above PoS tagging model we have planned to use the rule based (Transformation Based Learning (TBL)) and HMM by using Viterbi algorithm. Because many researchers developed PoS tagger in different language using hybrid approach and they showed better performance of tagger on different language. Hence, researcher will combined HMM with TBL (hybrid approach) to improve the performance of tagger.

2.4. Related Work

There are so many researches that have been done in the area of NLP in general and PoS tagging in particular for several languages of the world by using different approaches like rule based, transformation based learning, stochastic, artificial neuron network and hybrid approaches. Many researchers have been done on combining and comparing multiple algorithms for PoS tagging. Some of them are:

Part of Speech Tagging for Wolaita Language

In the study of [9] the author developed part of speech tagging for Wolaita language by using supervised and semi-supervised machine learning approaches like Conditional Random Field(CRF) and HMM using small manually tagged corpus. The researcher used 200

sentences which are manually tagged for training and testing purpose. The research result showed that HMM based taggers perform better than CRF based taggers.

Development of Part of Speech Tagger Using Hybrid

Another early work for Afaan Oromo[13] is developed by Getachew Emiru using hybrid approach. In his work, the researcher developed part of speech tagger using hybrid approach that combines rule based and HMM approaches conducted for Afaan Oromo language. The transformation based learner, which is a rule based tagger, tag the words based on rules, or transformations induced directly from the training corpus without human intervention or expert knowledge. The HMM tagger, tags the words based on the most probable path for a given sequence of words. For implementation and experiment, he used NLTK 3.0.2 and python 3.4.3 and 1517 sentences were used for training and testing, from these sentences 85% for training and the remaining 15% for testing. The performance analysis of the three taggers, namely: HMM, rule based and hybrid tagger were tested with the same training and testing set they achieved accuracy of 91.9%, 96.4% and 98.3%, respectively. Based on their performance and learning curve analysis, the study has concluded that the hybrid tagger has been benefited from the advantages of the two separated approaches and achieved an improved performance.

Design and Development of Part of Speech Tagger for Kafi-noonoo Language

The researcher developed part of speech tagger for Kafi-noonoo language in the paper of [11]. The author developed tagger using hybrid approach i.e. HMM and rule based tagger at sentence level. He used 354 untagged Kafi-noonoo sentence are collected from two genres and annotated using an incremental corpus preparation approach for training and testing purpose. The researcher identified 34 part of speech tags for tagging purpose, for training 90% of tagged sentence are used. The performance of HMM, rule based and hybrid taggers are tested using different experiment. As result, the performance of HMM, rule based and hybrid tagger shows 77.19%, 61.88% and 80.47% accuracy respectively. The author concluded that the hybrid tagger outperform for Kafi-noonoo language.

Hybrid Part-of-Speech Tagger for Vocalized Arabic Text

This research reported on the efficient and accurate part of speech tagging techniques for Arabic language using hybrid approach[14]. The HMM integrated with Arabic rule based method and to evaluate the accuracy of the proposed tagger, a series of experiment were conducted using holy Quran corpus and kalimat corpus for undiacritized classical Arabic language. The researchers used two corpuses to trained and tested PoS tagger for Arabic text. The authors were used the holy Quran corpus to evaluate PoS tagger , the evaluation rate are 97.6%, 96.8% and 94.4% for respectively hybrid tagger, HMM tagger and rule based tagger and the evaluation rate of kalimat corpus are 94.60%, 97.40% and 98% for respectively rule based tagger ,HMM tagger and hybrid tagger. The researchers concluded that accuracy of hybrid method represents a very good result compared with Tanni's rule based method and Albared's HMM method.

Part-of-Speech Tagger for Tigrigna Language

The research work in[17] developed PoS tagger for Tigrigna by using hybrid approach, HMM tagger combined with rule based tagger for Tigrigna part of speech tagger. As result, for training and testing purposes the author identified 36 broad tag sets and 26,000 words from around 1000 sentence containing 8000 distinct words were tagged. The researcher was used the way of combining HMM with rule based, first tagged raw Tigrigna text by using HMM tagger; afterward the rule based tagger is used as a corrector of HMM tagger. In this work the researcher was use Viterbi algorithm and Brill transformation based error driven learning are adapted for the HMM and rule based taggers respectively. The corpus divide into training set and testing set, for training set 75% of corpus was used and for testing set 25% was used. The different experiments are conducted for the three types of tagger (rule based tagger, HMM tagger and hybrid tagger). Thus, 89.13%, 91.8% and 95.88% performances are attained for HMM tagger, rule based tagger and hybrid tagger respectively. As result, the researcher concluded that the hybrid tagger is better than HMM tagger and rule based tagger used individually.

Towards Improving Brill's Tagger Lexical and Transformation Rule for Afaan Oromo Language

The aim of research attempt in [32] is to improve Brill's tagger and transformation rule for Afaan Oromo tagging with sufficiently large training corpus. The researcher identified and used 26 broad tag set and 1100 sentences (17,473 words) containing 6750 distinct words were tagged for training and testing purpose. Transformation based error driven learning are adapted for Afaan Oromo PoS tagger. The author used tenfold cross validation mechanism for testing the performance of tagger and compared with previously adapted brill's tagger, the previously adapted brill's tagger shows an accuracy of 80.08% whereas the improved brill's tagger result shows an accuracy of 95.6% which has an improvement of 15%.

Part-of-Speech Tagging For Afaan Oromo Language Using Transformational Error Driven Learning (Tel) Approach

The research work in [24] reported that the researcher develop part-of-speech tagger for Afaan Oromo using Transformational Error driven Learning (TEL) approach and compare it with other approach. The author used and identified 18 tag sets and used on 223 sentences (1708 words) for the experiment. The result found in the experiment is relatively good, 80.08% of the total word are correctly tagged. The result showed that Brill tagger was found to be better for Afaan Oromo than Hidden Markov Model.

Transformation-Based Part-of-Speech Tagging for Serbian language

The authors developed automatic part of speech tagger for Serbian language using transformation based learning in the paper [33]. The work described in this paper represents one of the first attempts at creating a completely automatic PoS tagger for Serbian language. The algorithm includes a language independent modification of the existing transformational-based approach so as to make it more convenient for languages with morphologically rich structure, particularly highly inflected languages. The researchers combined TBL tagging with stochastic tagging methods such as hidden Markov models (HMM) , HMM tagger as the initial tagger for TBL instead of assigning initial tags according to the relative frequency of tags in the training corpus and improved tagging accuracy and overcoming problems related to data sparsity for highly inflected languages.

Kadazan Part of Speech Tagging using Transformation-Based Approach

The research work in [34] researchers implemented the Part of Speech (PoS) Tagger for Kadazan language using Transformation-based approach and it also to solve the disambiguation problem in that language and at the same time to use it as a learning language tool. Authors used and implemented transformation based learning approach because it can achieve higher accuracy equivalent to other tagging approaches such as statistical and the original rule-based techniques and as well as having a significant advantages over the other tagging approaches.

The researchers trained the tagger using two Kadazan corpuses which contain 741 words and 1328 words. Based on the evaluation results, the tagging model can achieve around 92 % to 93 % of accuracy. By comparing corpus1 and corpus2 results for corpus1 obtained higher accuracy than corpus2. They concluded that tagging result for Kadazan language achieved high accuracy by using Brill's approach.

PoS Tagging for Marathi Language using Hidden Markov Model

In the work of Patil et al [35] addressed a problem of PoS tagging for Marathi language. The researcher used Supervised learning method that uses Hidden Markov Model is implemented to mark Marathi text using PoS tags. HMM model technique to train and test PoS tagger for Marathi and unigram, bigram and trigram language model used for experiment. The system is trained using 12,000 sentences and tested using 3000 sentences in Marathi language. The proposed system based on HMM algorithm has obtained satisfactory accuracy of 86.61% on test data.

Part-of-Speech Tagging For Afaan Oromo Language

The research work in [1] developed part of speech tagging for Afaan Oromo Language using HMM approach. The researcher has implemented unigram and bigram model of Viterbi algorithm and for training and testing purpose 159 sentences that are manually annotated sample corpus are used. The researcher was used Java programming language to develop the tagger prototype based on the Viterbi algorithm with unigram and bigram models. It is also used to compute both lexical probabilities and transitional probabilities. Tenfold cross validation mechanism used for testing the performance of tagger. The result shows that in unigram model 87.58% and in bigram model 91.97% accuracy is obtained.

A Hidden Markov Model for Persian Part-of-Speech Tagging

The research work in [36] the authors proposed PoS tagging system on Persian corpus by HMM and evaluate the accuracy of proposed approach, this proposed approach is applied in simulations which are done on both homogeneous and heterogeneous Persian corpus. High accuracy rate (98.1%) of two experiments (homogenous and a heterogeneous) part of the corpus demonstrates that Hidden Markov Models are suitable for PoS tagging in Persian language.

A Hidden Markov Model-Based PoS Tagger for Arabic

In [37] the researchers present part of speech tagger for Arabic language. The authors used stochastic approach that uses HMM to do PoS tagging of Arabic text and they used F-measure to evaluate the PoS tagger performance. The proposed HMM PoS tagger has been tested and has achieved a state of art performance of 97%.

Part-of-Speech Tagging of Urdu in Limited Resources Scenario

The research paper in[38] the researchers addressed the problem of PoS tagger of Urdu and they developed an automatic PoS tagger for Urdu text using different machine learning methods, namely HMM, maximum entropy model and conditional random field. They used 10,000 sentences manually annotated corpus. The research result showed that these three model to achieve higher accuracy than the existing models.

Hybrid Approach for part of Speech Tagger for Hindi Language

In [39] the researchers proposed hybrid based part of speech tagger for Hindi language. This system is developed using the combination of HMM tagger and rule based tagger and for the experiment the authors used 80,000 words corpus with 7 different standard part of speech tags for Hindi. This proposed system works in two ways-firstly input words are found in database, if it is present then it is tagged. Secondly if it is not present then applied various rule or HMM model. As result, the authors concluded that the hybrid approach has good performance for PoS tagging.

Hybrid Approach Based PoS tagging for Hindi language

In [40] the authors presented implementation of newer Hindi PoS tagger based on Hybrid approach (combination of Rule-based and Probability based) uses some linguistic resources,

several rules and database with some predefine list of possible prefixes, suffixes and other required and related data for the Hindi language.

Hybrid part of speech tagger for Malayalam

The research paper of[41] proposed an efficient and accurate PoS tagger for Malayalam language by using hybrid approach. Conditional Random Field (CRF) method integrated with rule based method and the researchers used SVM based method to compare the accuracy. Both tagged and untagged corpus used for training and testing the system and the proposed approach is Unicode based. As result, the authors presented the accuracy of 94% for PoS tagger of Malayalam.

Hybrid part of Speech Tagger for Sinhala Language

In the paper[42] the researchers developed part of speech tagger for Sinhala language using hybrid approaches. The authors combined stochastic with HMM approach and rule based approach, in this work the HMM model based stochastic tagger is constructed which is based on bigram probabilities then after addition rule based tagger was increase up the accuracy of tagger. In this research work the researchers concluded that the hybrid approach can be used to gain a higher PoS tagging accuracy for Sinhala language.

A Hybrid PoS Tagger for A Relatively Free Word Order language

In the research work of[43] the authors designed hybrid part of speech tagger for Tamil, a relatively free word order, morphologically productive and agglutinative language. In this work the researchers was use both combination of a HMM based statistical PoS tagger and a Rule based PoS tagger. The system works first, the HMM tagger trained using small corpus then given new sentence into tagger and they are tagged. There may be untagged word due to the limitation of algorithm and the amount of training corpus used. Those sentence or words which are not tagged given to rule based system and tagged. The authors used Viterbi algorithm for HMM tagger and to evaluate the accuracy of the taggers by using precision and recall. Hence, after analyzing the result of the tagger, they concluded that the hybrid is good accuracy for the correctness of tagged output.

Hybrid PoS Tagger

In [44] the author designed part of speech tagger for Romanian by using hybrid approaches. The researcher combined statistic model with rule based model. In this work, reducing the

tagging ambiguity by PoS dictionary before classifying the input word then tagging input word by using statistical tagger and finally correcting the error of statistical tagging by rule based system was attempted. Finally, the author presented the result of the model for Romanian hybrid tagger which is importantly enhancement of the tagging precision.

Table 1: Summary of Related Works

Authors	Journal types	Objective	Methodology
Berhanu Herano[9]	Addis Ababa University thesis	To investigate possibility of developing PoS tagger for Wolaita text using small manually tagged text	Supervised and unsupervised ML approaches (CRF, HMM)
Zelalem Mekuria[11]	Addis Ababa University thesis	To design and develop a PoS tagger for Kafi-noonoo language	Use hybrid approach(rule-based and HMM)
Getachew Emiru[13]	Addis Ababa University , thesis	To investigate the use of hybrid (rule based and HMM) approaches to the development of PoS tagging for Afaan Oromo	Use hybrid approach(rule based and HMM)
Teklay Gebregzabiher Aberha[17]	Addis Ababa University, thesis	To develop PoS tagger model for Tigrigna and analyze the performance of the model	Hybrid (HMM and transformation based learning based) approach used NLTK and Python programming language used for implementation of tagger
Kanak Mohnot, et al. [29]	International Journal of Computer Technology and Electronics Engineering	present a Hybrid Based Part of Speech Tagger for Hindi	Hybrid approach used i.e. HMM and rule based (hand writing rule) Precision, recall and F-measure are used for evaluation

	(IJCTEE), February 2014		
Getachew Mamo[1]	Addia Ababa, thesis, 2009	To investigate the possibility of designing and developing an automatic PoS tagger for Afaan Oromo language	HMM approach used Java programming language used to develop the tagger prototype based on the Viterbi algorithm with unigram and bigram models Tenfold cross validation used for testing performance of tagger
Fareena Naz et al. [45]	World Applied science journal, 2012	To present PoS tagger for Urdu language	TBL approach used Precision, recall and F-measure used for performance testing

Hence, it is observed from the review that many PoS tagging researches were done using rule based, stochastic, CRF, ANN and hybrid approaches for different languages. Some of the research works presented better accuracy using hybrid approaches rather than the individual method. So, in this research, a hybrid approach has been used to tag Wolaita language sentences. The approaches to be integrated for this experiment are transformation based learning and HMM method to develop PoS tagger for Wolaita language.

There are around four reviewed researches that are related with this research study; However, “PoS tagger for Kafi-noonoo Language” paper is the most related one. Because, the language is categorized on the Omotic family group and it uses Latin script. Besides of this, the researcher

has used hybrid approach as a method in order to develop a model which used to tag the Kafi-noonoo language sentences. These reviewed research indicates that the hybrid approach better performance than individual approaches. For the reason that, the implementation of hybrid approach on the Omotic family group can produce highly accurate output. Based on, the indication the researcher going to apply hybrid approach as a method in order to develop PoS tagger model to tag Wolaita Language sentences. The model developed for the Kafi-noonoo Language sentence tagger cannot be applied on the Wolaita Language because, they are morphologically different. As the indication of previously done research study on Wolaita Language PoS tagger model though using HMM and CRF method independently; however, the final output produced less accuracy because of the researcher has used small data. But, HMM approach requires large amount of data for better performance. Consequently, this research study follows hybrid approach as a method with relatively large corpus in order to improve the accuracy of tagger.

CHAPTER THREE

3. WOLAITA LANGUAGE

3.1. INTRODUCTION

This chapter identifies word categories for Wolaita Language for the development of PoS tagger. Wolaita language is one of the Northern Omotic languages that is spoken in the Wolaita Zone and some other parts of the Southern Nations, Nationalities, and People's Region of Ethiopia. The language has around 3.3 million native and dialective speakers [3]. Wolaita language is first written in 1940s by a team of missionaries led by Dr. Bruce Adam. They translated the bible into Wolaita in 2002[10]. Wolaita language serves as a medium of instruction in primary school students and is offered as subject for high school students in Wolaita zone. In addition, the Wolaita language department at Wolaita Sodo University offers bachelor degree in Wolaita language studies. The language is used to be written by using Amharic script (Fidel) before 1993. But after 1993, the language is being written using Latin script[9]. Several translations have been made so far from Amharic to Wolaita and from English to Wolaita using Amharic scripts as well as Latin script and most of the translation made are religious and materials.

Words are traditionally grouped into equivalence classes called part-of-speech (known as word classes, morphological classes, or lexical tags)[1]. For this study, the researcher identify word classes of Wolaita language sentences. For the purpose of the study, the researcher clarify and discuss word classes such as noun, verb, adjective, adverb, and preposition etc. of Wolaita language and identified tag set of Wolaita language which are used for the development of PoS tagger. In traditional(basic) grammars[1] there were generally only a few part-of-speech (noun, verb, adjective, preposition, adverb, conjunction, etc.)[28]. But, presently there are many more word classes additional for natural language processing in general and part-of-speech tagging in specific to identify words in sentences with their specific identities.

[1] Traditional grammar is outline for the description of the structure/syntax of language

3.2. Wolaita language Alphabet and Pronunciation

In Wolaita language there are 34 letters which are used to form words. These are written in two ways: capital and small letters.

Table 2: Wolaita Language Letters

Letter		Pronunciation In Amharic	Pronunciation in Wolaita	Letter		Pronunciation In Amharic	Pronunciation in Wolaita	Letter		Pronunciation In Amharic	Pronunciation in Wolaita
Capital	Small			Capital	Small			Capital	Small		
A	a	አ	Aa	M	m	ም	Mi	Y	Y	ይ	Yi
B	b	ብ	Bi	N	n	ን	Ni	Z	Z	ዝ	Zi
C	c	ጭ	Ci	O	o	ኦ	O o	CH	ch	ች	Chi
D	d	ድ	Di	P	p	ፕ	Pi	DH	dh		Dhi
E	e	ኤ	Ee	Q	q	ቅ	Qi	NH	nh	ኝ	Nhi
F	f	ፍ	Fi	R	r	ር	Ri	NY	ny	ኝ	Nyi
G	g	ግ	Gi	S	s	ስ	Si	PH	ph	ጵ	Phi
H	h	ህ	Hi	T	t	ት	Ti	SH	sh	ሽ	Shi
I	I	ኢ	Ii	U	u	ኡ	U u	TS	Ts	ፅ	Tsi
J	J	ጅ	Ji	V	v	ቭ	Vi	ZH	zh	ዥ	Zhi
K	k	ክ	Ki	W	w	ው	W i				
L	L	ል	Li	X	x	ጥ	Xi				

These Wolaita language letters are divided into vowels and consonants. In Wolaita language there are five kinds of different vowels. These are a, e, i, o and u. This letters are written in two ways, shortening and lengthening. When the word is shortening, we use a single letter of vowel like “a”, “e”, “i”, “o”, “u” and for the lengthening words we use double vowel like “aa”, “ee”,

“ii”, “oo” and “uu”. In Wolaita language, letters shortening and lengthening usage is different in meaning of word.

Akkaa -/dew/ Aaka -/be wide/

Awa- /sun/ Aawa -/father/

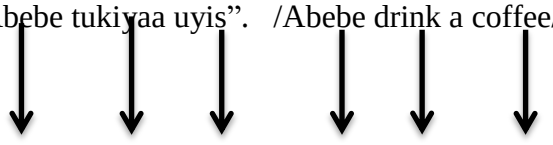
In the above Wolaita word the first word “Akkaa” means in English dew and “Aaka” means /be wide/ both have different meaning due to the lengthening of the vowel. In first word “Akkaa”-/dew/ the first letter “a” is shortened and in the word “Aaka”-/be wide/ the first letter “aa” doubled which is lengthening.

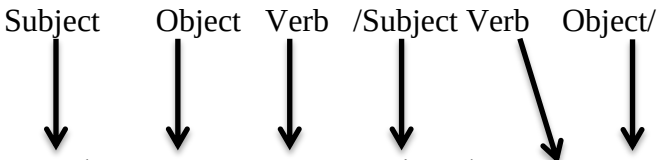
Consonants are letters those we pronounce with the vowels in a word. In Wolaita language there are twenty nine consonant letters. We read these letters in two ways. These two ways of reading are fastening and stressing. The consonants will be a single letter for words that are read with speed and double letters will be used for stressing the consonant in a word. For example the word “maataa”-/grass/ is read with speed or faster in Wolaita language word because the consonant “t” is single. But in the word “maattaa”- /milk/ is read by stressing the sound ‘t’.

3.3. Wolaita Language Sentence Structure

In any language, there are standard word orders in a sentence that are allowed to appear to convey a message. A word order is a syntactic method that shows relationship between words in a sentence. For instance, the word order of English and Franch is (SVO) subject – verb – object/counterpart. Whereas, in Amharic and Japanese languages, the order of words is subject – object – verb (SOV). As a result, the order of words in a sentence is a hint to label word categories as subjects and objects will be nouns and the action will have a verb class. Likewise, Wolaita language sentence is a set of words that contain subject, verb and object. The word order of Wolaita language is subject-object-verb (SOV). The following two Wolaita sentences represent the word order of the language.

“Abebe tukiyaa uyis”. /Abebe drink a coffee/.





“Maahe maataa meena”. - /A tiger doesn’t eat grass/.

In the above first sentence, the word “Abebe”- /Abebe/ is subject, “tukiya” /coffee/ is object and “uyis”-drink is verb, whereas in the second sentence the words “Maahe”-/tiger/,”maataa” -/grass/ and “meena”-/doesn’t eat/ are subject, object and verb respectively. Hence, the sentence structure in Wolaita language is SOV.

3.4. Word Categories

Due to the fact that words can be created by the speakers anytime, it is supposed that there are infinite set of words in a language. But not all words have the same role in a sentence, because some word express action, some designate a thing and other words join one word to another word. These are the building blocks of a language. When one wants to build a sentence, there is a need to use the different types of words with their varying role to convey the message. Each type of word has its own job. In order to determine the category of the word, linguists use morphological (internal structure of word), syntactic (structure of sentence) and semantic (meaning of a word) as a hint. While the task of part of speech tagging create many parts of speech that are used to create distinction between words with different roles and pronunciation, the basic list of word classes are small. The basic word classes in Wolaita languages are discussed below[10].

3.4.1 Noun Categories

Noun is part of speech that is used to describe person, things, organization, objects, etc. For example, the following shows the different types of nouns in Wolaita language:

Person: “Asa” /person/, “wondimu” /wondimu/

Place: “Amerka” /America/, “Adisaba” /Addis Ababa/

Things: “keettaa” /house/, “mayiyuwa” /clothe/, “mittaa” /wood/

Based on the suffixes, noun in Wolaita language is divided into four classes.

- ✓ The first class contains word that suffix **-a** in the absolute case and have stress on the last syllabus. For instance, such words are:

Table 3: The First Class of Nouns suffix 'a' Example

Wolaita language-Noun	English language-Noun	Wolaita language-Noun	English language-Noun
As**a**	Person	Xum**a**	Darkness
Deshsh**a**	Goat	Shuchch**a**	Stone
Qum**a**	Food	Aawa**a**	Father
Anijuwaa	Blessing		

The second class of nouns contains nouns that require their absolute case the word finishing –**iyaa/iya** and the stress on the syllable before the last one. For instance, such words are:

"Bitan**iyaa"** /married man/ " Kush**iyaa"** /hand/

"ayf**iyaa"** /eye/ "ged**iyaa"** /leg/

"maah**iyaa"** /tiger/

- ✓ The third noun class words finishing –**uwaa** in absolute case.

For example: such words are

"Met**uwaa"** /trouble / " poo'**uwaa"** /light/

"puutt**uwaa"** /cotton/ " ash**uwaa"** /meat/

"de'**uwaa"** /life/ "oos**uwaa"** /work/

- ✓ The fourth class noun ending –**iyo** in absolute case.

"Aayy**iyo"** /mother/ "Bollot**iyo"** /mather in law/

"Dottor**iyo"** /female doctor/ "asttamaar**iyo"** /female teacher/

"Tamaar**iyo"** /female student / "gawar**iyo"** /cat **(**female**)/**

In Wolaita language noun class do have four sub classes.

Table 4: Show Sub-Class of Noun in Wolaita Language

No	Sub-class	Example
1	Proper noun	Toophphiya /Ethiopia/, Sudaane- /Sudan/
2	Common noun	Deshsha /goat/, tuussaa /pillar/
3	Collective noun	Kata /Food/, Dozza /plant/
4	Abstract	Carikuwaa /wind/,erra/knowledge

3.4.2. Pronoun

In Wolaita language, a pronoun is defined as a word or phrase that is used as a substitution for a noun or noun phrase. Wolaita pronouns are also used in place of nouns as pronouns of other languages. As nouns, pronouns describe based on number and gender. For instance, “ne” - (you), “i” - (he), “a” (she), “nu” (we), etc.

Sub classes of pronoun

Table 5: Sub-class of Pronoun

No	Sub classes	Example
1	Personal pronoun	Ta/taani /I am/, nu/nunni /we/we/are/, ne/nenni /you/, intte, I /he/, a /she/, eti /they/
2	Demonstrative pronoun	Neega /yours/ iiga /hers/, nuugeeta-,taagaa, nu,nuussi
3	Interrogative pronoun	Aybee /what/ ,awugee /which/,ooni /who/, oogee /whom/
4	Indefinite pronoun	Amara /some/, issuwa /one/, ayibinne /nothing/, oonee /nobody/, ubbaa /everything/

3.4.3. Verb

Verbs are words that are used to express action or state of being. Verbs in Wolaita language exhibit very complex inflection system depending on mood, tense, kind of action and aspect[9]. For instance, if we take verb ekka /take/, it has the following inflection for past tense.

Ekkasi /I took/ *Ekkadasa /you took/*

Ekkasu /she took/ *Ekkis /he took/*

Ekkida /we took/ *Ekkideta /they took/*

Ekkidosona /they took/

Wolaita verbs are found at the end of the sentence just like Amharic language and suffix bound morphemes which help to indicate the subject of the sentence.

3.4.4. Adjective

An adjective is a word that describes, identify or quantify a word by preceding the noun or pronoun which it modifies. For instance, in Wolaita language words: “bootta” /white/, “zo’o” /red/, “qera” /small/, wogga /large/ and others are describing the nouns in the sentence of Wolaita language.

Examples for adjective:

Boota*/ADJ godariya/NN /white hyena/*

Wogga*/ADJ keetta/NN /large house/*

3.4.5. Adverb

An adverb is a word that tells us more about a verb; it modifies a verb, adjective and other adverbs. In Wolaita language modifiers of verb or verb phrase usually express time, location, manner, etc. for example:

*Abari **zino** yiis. /Abera came **yesterday/***

*Abari **eesuwan** yiis. /Abera came **quickly/***

In the first example, the word “zino” /yesterday/ used as adverb of time while in second sentence, the word “eesuwan” /quickly/ used as adverb of manner.

3.4.6. Preposition

In Wolaita language prepositions are words usually coming in front of noun or pronoun and express source, destination, location and relation to another word or element. For example:

*Tappe **simmin** - /**after** me/*

*Neeppe **kase** -/**before** you/*

*Algaappe **garssara** -/**under** the bed/*

3.4.5.7 Conjunction

Conjunctions are words that serve to connect words, phrase, clauses or sentence. For example: woyiko-/or/, gididogishshawu- /So/

3.5. Tags and Tag set of Wolaita Language

This section describe and identify the tags and tagsets used for the development of this study. Tag is a string used as a label in part of speech tagging. Tagset is a set of labeling string or collection of label that are identified and used for developing PoS of the study. In Wolaita language there is no readily accessible tag sets. The tag sets for the study are identified and developed in this section. Identifying and developing detail of tag set requires human experts and is time consuming. Hence, for this study, different category of tagset is identified based on basic word class of Wolaita language.

3.5.1. Noun Tag

In Wolaita language noun have different attributes like numbers, gender and case. Noun in Wolaita language can be proper noun, common noun, collective noun and abstract noun and it also singular, plural nouns. In this research work, common and proper nouns which have singular and plural cases and also collective and abstract nouns are represented by using the string NN .For instance:

Tophphee*/NN gitta* ***biitta****/NN /Ethiopia is a great country/*

Natti*/NN* ***keettaa****/NN giidoni dosona /****children's*** *found inside the* ***house****/*

Other type of noun rather than proper noun, common noun, abstract noun and collective nouns are: Noun with conjunction, noun with preposition, noun derived from verb and noun with adverb ending.

3.5.1.1. Noun with preposition:

In Wolaita language many nouns are not separated from proposition. There are eight prepositions in Wolaita language like English language prepositions: **u**- for, to, **yo**- for, to, **-kko**- toward, **ppe** from, **ra**- with, **ssi**- for, to, **ni**- in, at, by, and **danni**- like. For instance:

*Wolaitti Qolichcha****yo****/NP yeleta biitta /Wolaita is a birthplace* ***for*** *kolicha/*

3.5.1.2. Noun with conjunction

In English language noun do not come with conjunction. But in some Ethiopian languages like Amharic, Dawuro, Gamo, Wolaita, etc noun can come with conjunctions. In Wolaita language noun sometimes are not separated from conjunction. For instance, the word ending with "**nne**" means and, "**kkokka**"-means but and etc.

*Alemayoy****nne****/NC kabadi zino yidosona. /Alemayehu* ***and*** *kebede came yesterday/.*

3.5.1.3. Noun with verbal ending

In Wolaita language noun sometimes came with verbal ending. These are nouns formed by adding non autonomous words like –gaa, -gee to verb[9]. For this study noun with verbal ending are assigned by using the string of NV

For example: *nagiya****gee****/NV bawaa. /No guard man/.*

*He na'ay lo'iya****gaa****/NV. /That boy is handsome/.*

3.5.2. Pronoun Tag

In Wolaita language four sub class of pronouns, which are personal pronoun, demonstrative pronoun, interrogative pronoun and indefinite pronoun. Under this sub topic the string PP is used to represent personal pronoun, demonstrative pronoun, and indefinite pronoun. In the following example of Wolaita language sentence, the bolded words are represent pronouns.

Ta*/PP zino yasi. /****I*** *came yesterday/.*

Ne*/PP biiriya tawu ima. /give me* ***your*** *pen/.*

Neni*/PP ayiis yayayi? /Why* ***you*** *afraid?/*

Nussi*/PP xoossay lo''o. /God is good for* ***us****/.*

3.5.2.1. WH-Pronoun or interrogative expression tag

In this class, the word categories include **ayi** /what/, **awu** /where/ , **awuge** /which/ and **ooni** /who/. The interrogative word ***ooni*** -/who/ in this language is a personal pronoun. ***ayi***-/what/ is one of the most important interrogative words in this language. It is a widely-used interrogative (or indefinite) word for a nominal that refers to one or more non-animate things. It is invariable. It can be used like an absolute nominal. **Awu**-/where/ is indeclinable. ***Awu***-/where/ can be used adverbially by itself. In this work WH-pronoun are assigned WHP. For example:

Table 6: Example for WH-Pronoun/Interrogative Expression Tag

Wolaita language sentence	English sentence
Ooni/WHP yiidi de'i?	**Who** is coming?
Awu/WHP bayi neni?	**Where** are you going?
Ayi/WHP ootayi neni? **Ayi** be'ádii?	**What** are you doing? **What** did you see?
Awuge/WHP ne keehippe dosiyo qumay?	**Which** one is your favorite food?

3.5.2.2. Pronoun with conjunction

In English language pronoun do not come with conjunction. But in some Ethiopian languages like Amharic, Dawuro, Gamo, Wolaita, etc pronoun can come with conjunctions. For this study pronoun with conjunction are assigned using the string of PC. For example:

*Taani**nne**/PC ta'ishay issippe boos. /Me **and** my brother go together/.*

3.5.2.3. Pronoun with preposition

In English language pronoun are not come with preposition. But in some Ethiopian languages like Amharic, Dawuro, Gamo, Wolaita, etc pronoun can come with preposition. For this study pronoun with preposition is assigned using the string of PPRP

For example:

*Nuupe**kka**/PPRP nu**yo**/PPRP xoossi lo''o. /God is good to us **from** us/.*

3.5.3. Verb Tag

In Wolaita language seven sub class of verbs according to wakasa research work of a sketch grammar of Wolaita[46]. Which are: imperfective, perfective, future, optative, peripheral main verb endings, subordinate verb ending and verb infix. But for the purpose of this research work, the work of Berhanu[9] verb classification is used. Those are: main verbs, subordinate verbs, relative verbs and infinitive verbs.

3.5.3.1. Main verb tag

The main verbs include imperfective, perfective, future and optative verbs. The ending of the main verbs in Wolaita language are **ays, aassa, ees, awsu, oosona, iyo, eetii, eeta, ay, ii, oos, ikkii, ikke, okko, ettenna, ukku, akka, nna, iyode** and others. For instance, of main verbs below in the sentence:

Table 7: Example for Main Verb Tag

Wolaita language sentences	English sentences
Zino neeni **yiyode/**VV tanni katta **maays**/VV.	When you **came** yesterday, I was **eating** food.
Nenni minna oottikko lo''osa **gakkanna**/VV.	If you work hard, you will **reach** good place.
Nenni ayis **siyikkii**/VV.	Why don't you **hear**?

3.5.3.2. Subordinate verbs tag

In Wolaita language verbs can come with conjunctions or preposition. The verbs that are not separated form conjunction and preposition are included in this verb sub class. For this study verb with conjunction or position is assigned using the string of VC. For instance, some this type of verb below in the sentence.

*A **tamaaradanne**/VC erada diccasu. – /She was grown by **education and** understanding/.*

Zaaruwa **gigissi**/VC wottiis. – /He had **prepared** and put an answer./

3.5.3.3. Relative verb tag

Relative verb modifies the next word. Most of the time it come before noun in Wolaita language. Examples of relative verb are:

*Taani **shammido**/VR kettay lo''ii? - /Is it good the house that I **bought**?/*

*Taani issippe **yido**/VR na'a. – /The boy **came** together with me/*

3.5.3.4. Infinitive verb tag

In Wolaita language, infinitives are formed regularly from actual verb stem[9].

Example of infinitive verb given below in the sentence:

*Ha oogiyaa **kantanawu**/VI metees. – /It is difficult to **across** this road./*

3.5.4. Adverb tag

An adverb is a word that expresses more about a verb. It describes a verb, adjective and other adverbs. In Wolaita language Adverbs comes before a verb and it used to modify verb or verb phrase usually express time, location, manner, etc.

*Abari **zino**/ADV yiis. – /Abera came **yesterday**./*

*Abari **eesuwan**/ADV yiis. – /Abera came **quickly./***

The above bold word represent adverb of Wolaita language which describe/express time.

3.5.4.1. Adverb with conjunction or preposition tag

In English language adverbs do not come with conjunction or preposition. But in some Ethiopian languages like Amharic, Dawuro, Gamo, Wolaita, etc adverbs can come with conjunctions or preposition. For this study adverb with conjunction or position assigned by using the string of ADVC. For instance:

*Sa'ay zino**nne/ADVC** hachchi/ADV lo'o. – /**Yesterday and** today the condition is good./*

3.5.5. Adjective tag

In Wolaita language, adjectives modifies noun. Adjectives in Wolaita language differ from nouns in the following respects[10]: 1) Adjectives have not been observed to inflect for case, definiteness, number, or gender like nouns do. 2) Adjectives may be modified by an intensifier, but the intensifier ***keehi*** /very/ has not been observed modifying a noun directly. 3) While adjectives might be considered to be a closed class of words in that there must be fewer items in it than nouns, it is presumed that new adjectives could be included, which would make it an open class of words. Adjective in Wolaita language end in '**a'**, '**iya'**, '**uwa'**,**'o'**,. the following are example of Wolaita language adjectives.

*Adjective end **'a'***	*Adjective ends **'iya'***	*Adjective ends **'uwa/o'***
Geshsha- /clear	***Lo'iya- /Good/***	***Lo'uwa- /Nice***
Qanita - /Short/	***Karexxiya- /Black***	***Lo'o- /Good/***
Adussa - /long/	***Maa'iya- /sweet/***	

We have used the string **ADJ** to represent normal adjective. For example:

Abari/NN **lo'o**/ADJ asaa/NN. – /Abera is a **good** person./

Kareetta/ADJ godariyaa –/**black** hyena /

3.5.5.1. Adjective with conjunction or preposition

In English language adjective are not come with conjunction or preposition. But in some Ethiopian languages like Amharic, Dawuro, Gamo, Wolaita, etc adjective can come with conjunctions or preposition. For this study adverb with conjunction or position assigned by using the string of ADJC.

For example:

*Kareetta**nne**/ADJC botta/ADJ godariyaa- /Black **and** white hyena/*
*Abari lo'o**nne**/ADJC iita/ADJ.- /Abera Good and Bad/z*
*Adussa**ppe**/ADJC qanita/ADJ ekka. -/**from** long take short*

In the above words 'adussa**ppe**' represent adjective with preposition. the suffix '**ppe'**/from/ represent preposition and the suffix '**nne'** in word' lo'o**nne**'-/Good and/ represent conjuction.

3.5.6. Conjunction tag

Conjunctions are words that serve to connect words, phrase, clauses or sentence. For instance, the sentence with conjunction in Wolaita language is given below.

*Abebe ta lagiyaa **woykko**/CJ tayisha giddena. -/ Abebe is not my friend **or** my brother./*

*Qolchi tappe kasse **woykko**/CJ tappe simmin yiis- /kolcha came after **or** before me./*

3.5.7. Preposition tag

Prepositions in Wolaita are nominal. Wolaita, nominal that modify prepositions are in the oblique, as nominal that modify other nominal. Prepositions in Wolaita may change their forms depending on the context. There are eight postpositions in Wolaita: -***u***-/for/, /to/, -***yyo***-/for/, /to/', -***kko***-/toward/, -***ppe***-/from/, -***ra***-/with/, -***ssi***-/for/, /to/', -***ni***-/in/, /at/, /by/', and –***daani***-/like/.

Non-predicative	Predicative	Interrogative

for, to	u	*	*
for, to	-yyo	*	*
toward	-kko	-kko	- kkoo
from	- ppe	- ppe	-ppee
with	-ra	-ra	-ree
for, to	-ssi	-ssa	-ssee
in, at, by	-n(i)	-na	-nee
like	-da(a)n(i)	-daana	-daanee

For instance, from the above preposition **daani**-/like/ used to express degree. Mainly used to present an alternative view of a thing or situation and used to mention similar things.

The original use of the postposition –***ra-***/with/ appears to be to express whatever is in the very neighborhood of something. It is typically used when things exist side by side. The preposition –***ppe-***/from/ can express potential sources in space. The use of the preposition –***kko-***/toward/ appears to be to denote direction toward which something moves. Arrival at a destination is not necessarily implied. That is, referents of objects of the postposition usually serve just to determine or explain direction. They are not usually destinations to be reached.

Functionally, non-predicative forms of prepositions roughly correspond to absolutive or oblique forms of other nominals, and predicative forms of postpositions to absolutive forms of other nominals. Interrogative forms of prepositions are found at the end of prepositional phrases that serves as predicates of affirmative direct interrogative clauses [10]. This tag represented to PRP. The other sub-class of preposition in Wolaita language is preposition with conjunction which is tagged by PRPC.

For instance,

*Neeni biidoyi oonaa**ree**? - /Who is it that you went **with**?/*

*I diyoyi areka**nee**? - /Is it **in** Areka that he lives?/*

*Bodite sooddo**ppe** haakkées.- /Boditi is far **from** Sodo./*

*Kayisóyi ushacha**kko** bíis. - /The thief went to the right./*

*Qolichchi tappe **kasse**/PRP woykko/CJ tappe **simmin**/PRP yiis – /Qolcha come before or after me./*

*Tappe simmini**nne**/PRPC tappe kasse- / before and after me*

3.5.8. Interjection tag

In Wolaita language, interjections are words that are used to express emotion or strong feeling, to address or respond to the hearer or to fill blanks in a sentences caused by various reasons[10]. For example:

xooso tana ekka!- /OMG!/

Paa!/INT ayiba lo''I - /wow! its good/

3.5.9. Numerals tag

Wolaita language numerals like that of English, Amharic and Afaan Oromo can be cardinal or ordinal. In Wolaita language cardinal numerals are nearly working in place of adjective. Hence, it is considered under adjective tagset and assigned AJN. The other type of numerals in the language is ordinal numerals and all assigned by ON. In English language numerals are not come with conjunction or preposition. But in some Ethiopian languages like Amharic, Dawuro, Gamo, Wolaita, etc numerals can come with conjunctions or preposition. For this study numerals with conjunction or preposition assigned by using the string of CNN. For instance:

Oyidantta/ON- /fourth/ ***hezzu**/AJN laytta- /**three** years/*

*Issuwape**nne**/CNN naa'a**ppe**/CNN gidduwan- /**between** one **and** two/*

3.5.10. Demonstrative Determiners tag

According to wakasa[10], there are five demonstrative determiners in Wolaita language that help as bases in various demonstrative expressions. They are: **ha**-this, **he**- /that/, **hi**- /that/, **ha**-/in the nearest place/ and **ya**- /in the remote place/. The demonstrative determiner ***ha*** /'this'/ is widely used for proximal demonstrative expressions. It modifies a nominal, irrespective of number, gender, and case. *Yá-/*in the remoter place'/ is also used for distal demonstrative expressions. Unlike the distal demonstrative determiners, however, it is used to refer to what is in the remoter place between

the two (or occasionally more than two) things or persons that can be identified by both the speaker and the hearer. The demonstrative determiner ***he-***/that/ is widely used for distal demonstrative expressions. Thus it is a semantic complement of *ha-*/this/. This tag assigned DD. For instance:

Ha/DD bitanne zino de'es- /**This** man was attended yesterday./

3.5.10.1 Demonstrative Determiners with preposition tag

Demonstrative determiners in Wolaita language sometimes came with preposition or conjunction. In this study the same string DPRP was used to represent tag of demonstrative determiner with preposition or conjunction. For instance:

*Happe**nne** ya**ppe** awugee lo'o?- / Which is Good from this and that /*

3. 5.11. Punctuation tags

All forms of punctuation in Wolaita language are represented by using the string PUN. Such punctuations includes **?, . ,** ! and others.

This chapter has identified and described the word class of Wolaita language. In Wolaita language there is no tag set prepared for the purpose of NLP but there is a few research works on grammar for Wolaita language by different computational linguistic experts. After reviewing different literatures on grammars of Wolaita language, rules and word classes, the researcher identified the following tag set for the purpose of this research. Most tagsets in this research work is based on the work of Wakasa (2008) and work of Berhanu H. on Wolaita language word class; he[9] used 22 tagsets in his work. In this thesis, after understanding language feature and discussing with language expert, 26 tag sets were identified for the purpose of this study. The following table shows the summary of tagsets that are used for this study.

Table 8: Tagset Summary

No	Basic class	Derived class	Description	Examples
1	Noun	NN	All common and proper nouns singular and plural tag	***Tophphee**/NN gitta **biitta**/NN /Ethiopia is a great country/*

2		NP	Noun not separated form preposition tag	*Wolaitti Qolichcha**yo**/NP yeleta biitta /Wolaita is a birthplace **for** kolicha/*
3		NC	Noun not separated form conjunction tag	*Alemayoy**nne**/NC kabadi zino yidosona. /Alemayehu **and** kebede came yesterday/.*
4		NV	Noun with verbal ending	*He na'ay lo'iya**gaa**/NV. /That boy is handsome/.*
5	Pronoun	PC	Pronoun with conjunction	*Taani**nne**/PC ta'ishay issippe boos. /Me **and** my brother go together/.*
6		PPRP	Pronoun with preposition tags	*Nuupe**kka**/PPRP nu**yo**/PPRP xoossi lo''o. /God is good to us **from** us/.*
7		WHP	All Interrogative pronoun tags	**Awuge/**WHP ne keehippe dosiyo qumay? /**Which** one is your favorite food? /
8		PP	All personal pronoun	***Nussi**/PP xoossay lo''o. /God is good for **us/**.*
9	Verb	VV	All main verb tags	Nenni minna oottikko lo""osa **gakkanna**/VV. /If you work hard, you will **reach** good place. /
10		VI	All infinite verb tags	*Ha oogiyaa **kantanawu**/VI metees.*
11		VC	All subordinate verb including conjunctions and preposition tags	*A **tamaaradanne**/VC erada diccasu. – /She was grown by **education and** understanding/.*
12		VR	All relative verb tags	*Taani **shammido**/VR kettay lo''ii? - /Is it good the house that I **bought**?/*
13	Adverb	ADV	All adverb tags	*Abari **eesuwan**/ADV yiis. – /Abera came **quickly./***
14		ADVC	Adverb with conjunction or preposition	*Sa'ay zino**nne/ADVC** hachchi/ADV lo'o. – /**Yesterday and** today the condition is good./*

15	Adjective	ADJ	All adjective tags	**Kareetta**/ADJ godariyaa –/**black** hyena /
16		ADJC	Adjective with conjunction or preposition tags	*Adussa**ppe**/ADJC qanita/ADJ ekka. -/**from** long take short*
17	Numerals	AJN	All cardinal numeral tags	***hezzu**/AJN laytta- /**three** years/*
18		ON	All ordinal numeral tags	*Oyidantta/ON- /fourth/*
19		CNN	Cardinal with conjunction tags	*Issuwape**nne**/CNN naa'a**ppe**/CNN gidduwan- /**between** one **and** two/*
20	Preposition	PRP	All preposition tags	*Qolichchi tappe **kasse**/PRP woykko/CJ tappe **simmin**/PRP yiis – /Qolcha come before or after me./*
21		PRPC	Preposition with conjunction tags	*Tappe simmini**nne**/PRPC tappe kasse- / before and after me*
22	Conjunction	CJ	All conjunction tags	*Abebe ta lagiyaa **woykko**/CJ tayisha giddena. -/ Abebe is not my friend **or** my brother./*
23	Interjection	INT	All interjection tags	***Paa!**/INT ayiba lo''I - /**wow!** its good/*
24	Determinant	DD	All determinant tag	**Ha**/DD bitanne zino de"es- /**This** man was attended yesterday./
25		DPRP	Determinant with preposition tag	*Happe**nne** ya**ppe** awugee lo'o?- / Which is Good from this and that /*
26	Punctuation	PUN	All punctuation tag	**?, . ,** !

CHAPTER FOUR

4. MATERIALS AND METHODS

4.1. INTRODUCTION

A research methodology defines what the activity of research is and how to proceed. In this thesis the researcher used design science research method to follow to design PoS tagger model for Wolaita language. Design science is outcome based information technology research methodology. Data science is an inter-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from many structural and unstructured data. The Data Science Methodology is an iterative system of methods that guides data scientists on the ideal approach to solving problems with data science, through a prescribed sequence of steps[47].

The following steps are generally used in design science research. These are list and described by[48].

i. From Problem to Approach

No work start without motivation, Data science is no exception though. It's really important to declare or formulate your problem statement very clearly and precisely. Your whole model and its working depend on your statement. Many scientist considers this as the main and much important step of Date Science. So make sure what's your problem statement and how well can it add value to business or any other organization. Once the business problem has been clearly stated, the data scientist can define the analytic approach to solving the problem. This stage entails expressing the problem in the context of statistical and machine-learning techniques, so the organization can identify the most suitable ones for the desired outcome.

ii. From Requirements to Collection

The chosen analytic approach determines the data requirements. Specifically, the analytic methods to be used require certain data content, formats and representations, guided by domain knowledge.

In the initial data collection stage, data scientists identify and gather the available data resources—structured, unstructured and semi-structured—relevant to the problem domain.

Typically, they must choose whether to make additional investments to obtain less-accessible data elements. It may be best to defer the investment decision until more is known about the data and the model. If there are gaps in data collection, the data scientist may have to revise the data requirements accordingly and collect new and/or more data. After the original data collection, data scientists typically use descriptive statistics and visualization techniques to understand the data content, assess data quality and discover initial insights about the data. Additional data collection may be necessary to fill gaps.

iii. From Understanding to Preparation

This stage encompasses all activities to construct the data set that will be used in the subsequent modeling stage. Data preparation activities include data cleaning (dealing with missing or invalid values, eliminating duplicates, formatting properly), combining data from multiple sources (files, tables, platforms) and transforming data into more useful variables. Data preparation is usually the most time-consuming step in a data science project. In many domains, some data preparation steps are common across different problems.

iv. From Modeling to Evaluation

Starting with the first version of the prepared data set, the modeling stage focuses on developing predictive or descriptive models according to the previously defined analytic approach.

With predictive models, data scientists use a *training* set (historical data in which the outcome of interest is known) to build the model. The modeling process is typically highly iterative as organizations gain intermediate insights, leading to refinements in data preparation and model specification. For a given technique, data scientists may try multiple algorithms with their respective parameters to find the best model for the available variables. During model development and before deployment, the data scientist evaluates the model to understand its quality and ensure that it properly and fully addresses the business problem. Model evaluation entails computing various diagnostic measures and other outputs such as tables and graphs, enabling the data scientist to interpret the model's quality and its efficacy in solving the problem. For a predictive model, data scientists use a testing set, which is independent of the training set

but follows the same probability distribution and has a known outcome. The testing set is used to evaluate the model so it can be refined as needed. Sometimes the final model is applied also to a validation set for a final assessment.

v. From Deployment to Feedback

Once a satisfactory model has been developed and is approved by the business sponsors, it is deployed into the production environment or a comparable test environment. Usually it is deployed in a limited way until its performance has been fully evaluated. Deployment may be as simple as generating a report with recommendations, or as involved as embedding the model in a complex workflow and scoring process managed by a custom application. Deploying a model into an operational business process usually involves additional groups, skills and technologies from within the enterprise. By collecting results from the implemented model, the organization gets feedback on the model's performance and its impact on the environment in which it was deployed.

4.2. Methodology of study

Designing the correct Research methodology is very important to do any scientific research. In this research, the following methods were used to achieve the goal of the research. These are literature review, data collection, data preprocessing, design and implementation and test and evaluation.

4.2.1. Literature review

Reviewing of the relevant literatures conducted to gain deep understanding of the research area. To have a better and solid understanding of the problem domain, reading of books that related with my study, journals and research articles relevant to the research topic was undertaken. In this research work, literature review is very important phase to understand the word class and the morphological property of Wolaita language. In this strategy to review the literature in the area of part-of-speech tagging based on HMM model and rule based model and discuss with linguistic and expert for more understanding of Wolaita language.

4.2.2. Data collection

The required data collected for this study was gathered from different sources. These are online bible in Wolaita language[49] (30%), social media in Wolaita language (Wogetta FM 96.6)(40%), and also different reference materials of Wolaita language which are provided by the department of Wolaita language in Wolaita Sodo University(30%). From these three data sources the researcher collected 1134 sentences or 14,358 words. The dataset used in this study is identified and prepared by the help of language expert who are mentioned in the next section. Moreover, this research is motivated by the recommendation of pervious work by Birhanu.H[9]. Based on the ideas of those experts the researcher identified 26 tagset to develop PoST for Wolaita language.

4.2.3. Data preprocessing

Data preprocessing is an important step in any scientific research work. In this phase collected data are preprocessed by removing foreign words[2], quotation marks and correcting spelling errors before annotating the corpus. Annotation of corpus was done manually with help of language experts after preprocessing. The collected corpus was manually tagged by the help of Mr. Marikose Ajaja and Mr.Kuleno Kolichaa. Mr. Marikos Ajaja is a teacher and, author for grade six text book of Wolaita language and graduate student in English department. And also Mr. Kuleno Kolichaa is a secondary school Wolaita language teacher in Areka secondary and preparatory school. After tagging it manually, the entire corpus were divided into sub corpus as 90% (1020 sentences or 12,683 words) for training dataset and 10% (114 sentence or 1,675 words) for testing dataset. Totally, the preprocessed data for this study is 1,134 sentences or 14,358 words. The sample of training and test dataset is put in appendix **A** at the end of this document. In this study two way data preprocessing is used: the first data preprocessing is done before preparing annotated corpus and second data preprocessing step in this study is after preparing annotated corpus to preprocess by using system to segment and to tokenize before giving the data for the learning algorithm.

[2] Any language except Wolaita language.

The challenging feature in Wolaita language POS tagging is the complexity of the morphological features in WL. Even some parts of speech are attached with other part of speech. Preposition, conjunction are usually attached with other part of speech. The other challenging feature in WL PoS tagging is, in Wolaita language quotation mark is used for two purposes, one purpose is as a quotation mark and the second usage is using as a character to form word . So, this a big challenge when the researcher preprocess the given corpus.

4.2.4. Design and implementation

Many algorithms have been applied for part of speech tagging including hand-written rules(rule-based tagging), probabilistic methods (HMM tagging and maximum entropy tagging) and Artificial neural network, as well as other methods such as transformation based tagging, memory-based tagging and combination(hybrid) approaches[50]. The research design combines two approaches. The approaches used here are the rule based (Transformation Based Learning (TBL)) and HMM by using Viterbi algorithm. From HMM, rather than using the built-in n-gram function, Viterbi algorithm was used. Rule based part of speech tagger is an approach that solves the problem of assigning the part of speech tags to words in sentences using rules extracted from language experts based on the morphemes attached onto words. These rules can be manually prepared by linguistic professionals and machine learned rules or transformation based learning. Transformation based learning is machine learning technique to generate rule by comparing manually tagged corpus with temporary corpus which is tagged by initial tagger; Transformational based learning takes un annotated corpus as input which goes through the initial state tagger. This initial state tagger assigns a tag that is most likely. This initial tagger produced a temporary corpus as an output then the temporary output corpus by the initial state tagger compared with the goal corpus which was manually tagged and expected to be correct. The corpus passes through the learner iteratively to derive rule for transformations. Each of these derived rules is examined by applying it to the temporary corpus and comparing the result with the goal corpus. Based on this comparison the highest score is applied to the text and is produced as ordered list of rules and added to the result list. The process continues until temporary corpus match with goal corpus or no change in rule.

In this research work, the machine learned rules used rather than using handcrafted rules because which consumes more time and needs skilled linguistic professionals. Machine learned

rules can be obtained on the course training of tagger. That means a model is made to automatically learn and store rule called brill Transformation from the training corpus to be provided[12]. Transformation-based learning (TBL) is a rule-based algorithm for automatic tagging of parts-of-speech to the given text[11].Transformation based learning transforms one state to another using change rules to find the appropriate tag for each word.

From the stochastic approach the hidden Markova model (HMM) tagger was designated to tag the words based on the most probable path of the word on a given sentence.

4.2.4.1. Hidden Markov Models

HMM is the most widely used technique for part of speech tagging in stochastic approach[51]. It is the probabilistic function of Markov process, a process which moves from state to state, left to right on the states, to find optimal state sequence[17]. A Hidden Markov Model (HMM) allows about both observed Model events (like words that we see in the input) and hidden events (like part-of-speech tags) that we think of as causal factors in our probabilistic model[13].

Training of the HMM Tagger

This research work implemented HMM tagger by using Viterbi algorithm. This algorithm perform ML to minimize the complexity of HMM tagger in terms of time and memory requirement. The Viterbi algorithm finds the best tag sequence without explicitly computing all sequences.

To find the best tag sequence for an entire sentence, the method uses the following formula[28]:

$$\hat{T} = \underset{T}{argmax}\, P(T/W) \qquad (4.1)$$

Where

$\hat{T}$ is sequence of tag

T is the most probable tag sequence

W is the sequence of word in the sentence

$\underset{T}{argmax}\ P(T/W)$ is a function given by T such that P(T/W) is largest

According to Bayes" law [5], P(T/W) can be expressed as:

$$\mathrm{P(T/W)} = \frac{\mathrm{P(T)P(W/T)}}{\mathrm{P(W)}} \tag{4.2}$$

Based on this, Equation 4.1 can be rewritten as:

$$\hat{\mathrm{T}} = \underset{T}{argmax}\ \frac{P(T)P(W/T)}{P(W)} \tag{4.3}$$

Since we are looking for the most likely tag sequence of a sentence given a word sequence, the probability of the entire sentence i.e. P(W) has a constant value for each tag sequence within the sentence and we can ignore it. As a result Equation 4.3 can be rewritten as:

$$\hat{\mathrm{T}} = \underset{T}{argmax}\ P(T)P(W/T) \tag{4.4}$$

According to the chain rule of probability[28], P(T)P(W/T) can be computed as:

$$P(T)P(W/T) = \prod_{i=1}^{n} P(w_i/w_1 t_1 \dots w_{i-1} t_{i-1} t_i)\, P(t_i/w_1\, t_1) \dots w_{i-1} t_{i-1} \tag{4.5}$$

Where wi is the ith word in the given sentence and ti is the ith tag in the tagset.

Due to too many possible sentences, it is difficult to compute the chain rule probabilities through count and divide mechanism. In order to compute these probabilities, we make the following assumptions[11].

1. The probability of a word is dependent only on its tag (Markov simplifying assumption).
2. Tag history can be approximated by the most recent tag

Based on these assumptions, Equation 4.4 can be rewritten as:

$$\hat{\mathrm{T}} = \underset{T}{argmax} \prod_{i=1}^{n} P(t_i/t_{i-1})\, P(w_i/t_i) \tag{4.6}$$

The first factor of this product is called lexical model and the second factor is called the contextual model.

The lexical model computes lexical probabilities of each word in the training data. In order to calculate these probabilities, we use maximum likelihood estimation from relative frequencies using the following formula[13].

$$P(w_i/t_i) = \frac{C(w_i,t_i)}{C(t_i)} \quad (4.7)$$

The contextual model which is also called the N-gram model computes contextual probabilities (information about the context where the word is found in the given sentence). These can be achieved by maximum likelihood estimation from relative frequencies as[28]:

$$P(t_i/t_{i-1}) = \frac{C(t_{i-1},t_i)}{C(t_{i-1})} \quad (4.8)$$

By changing the value of n, we can generate different N-gram model. Most of the time, in POS tagging development, n is equal to 2 or 3. The choice of value for n is dependent on the training data. Bi-gram model is preferable for large tagset or small training data.

The Viterbi Algorithm

The Viterbi algorithm is widely used in the NLP world that allows considering all the words in the given sentence simultaneously and computes the most likely tag sequence. The applicability of the Viterbi algorithm as presented by Jurafisky and Martin[28] is explained as follows: For any model, such as an HMM, that contains hidden variables, the task of determining which sequence of variables is the underlying source of some sequence of observations is called the decoding task. The Viterbi algorithm is decoding perhaps the most common Viterbi decoding algorithm used for HMMs, whether for part-of-speech tagging or for speech recognition[50].

The term Viterbi is common in speech and language processing, but this is really a standard application of the classic dynamic programming algorithm, and looks a lot like the minimum edit distance algorithm[13]. This research work implemented HMM tagger by using Viterbi algorithm. This algorithm perform ML to minimize the complexity of HMM tagger in terms of time and memory. The algorithm finds only the best path of each node at each position in the sequence and discards the others. The Viterbi algorithm finds the best tag sequence without explicitly computing all sequences.

Design of Wolaita language HMM Tagger

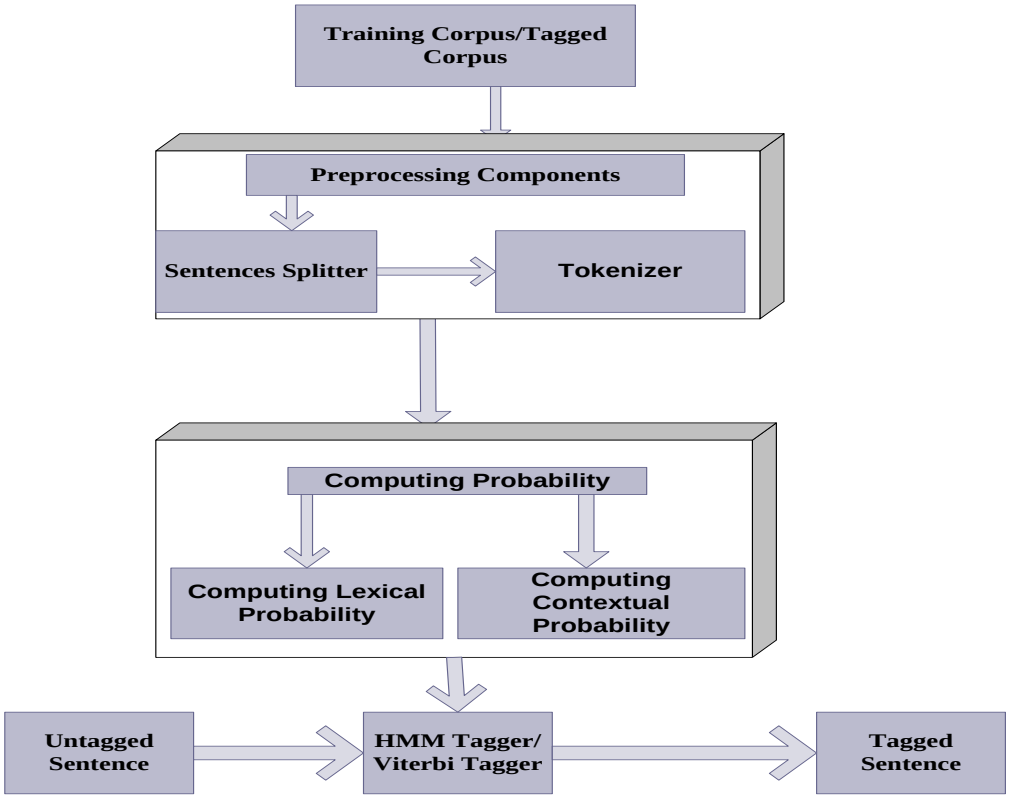


Figure 3: Design of HMM Tagger

Description of process in developing model is briefly described below:

i. Tagged corpus

Corpus is most important part of natural language processing (NLP) task like part of speech tagging. Corpus may be in two different forms: untagged and tagged or annotated corpus. For the purpose of this study, the researcher collected data from three domains and preprocessed the collected data. Annotation of corpus is tagging process adding linguistic information to an electronic corpus of written or spoken language data. Tagged corpus is a collection of textual data that contains linguistic information. Whereas untagged corpus is a collection of text without linguistic or grammatical information or untagged words.

Tagged corpus can be used for various purposes. In linguistics, properly tagged corpus can be used to study linguistic features such as morphology and phonology of a language. It can be used as part of speech taggers and parsers training in computational linguistics. Hence, annotated or tagged corpus plays a great role in automatic part of speech tagging and parsing[13].

ii. Preprocessing components

In this study, preprocessing components have three major parts; sentence splitter, tokenizer and tagset analyzer. A processed corpus is input for the model. At first, the content of corpus is given to sentence splitter module to split the text into sentence by using the Wolaita language sentences end markers such as **'.**','**?**'. The splitted sentence is given to tokenizer module to split the string into words and punctuation. During training phase tagged words were tagged in the form words/tags. The tagset analyzer extracts the tags from the output of the tokenizer. This extracted tagset is used for HMM tagger to tag a new text[52].

iii. Computing probability

Computing lexical and contextual probability from training set; lexical probability from training data set by count the frequency of word with given tag divided by total number of word in training data set. Lexical probability contains the likelihood probability of the word in training corpus and contextual probability contains the transition probability of the word in the training set. These probabilities are given to Viterbi model to pick the optimal path of the word from the two models (lexical and contextual probability). Finally the Viterbi matrix analyzes the most probable path and assign the word with its tags and tag sequence generator generate the word with assigned tags.

iv. HMM Tagger

HMM tagger is the main class of the tagging program/model. It takes untagged sentences as input and produces a tagged sentences or text as output. For instances:

*" **Neeni tanaara de'iyoogan taani daro ufayittas. "***

The tagger take the above sentence as input then to tag words using extracted tag in training time based on the best path of word to produce tagged sentence as output. The following sentences as output of tagger.

"Neeni/PP tanaara/PPRP de'iyoogan/VR taani/PP daro/ADV ufayittais/VV./PUN"

4.2.4.2. Transformation Based Learning

Transformation based learning is an algorithm that automatically extracts rule or linguistic information from a sample of manually annotated corpus and learns lexical or morphological and contextual information from correctly annotated corpus without human intervention or expert knowledge. It only requires a small manually correctly annotated corpus. The system is then able to derive lexical and contextual information from the training corpus and likely PoS tag for a word. Once the training is completed, the tagger can be used to annotate new, untagged corpus based on the tagset of the training corpus. Transformation based learning (TBL) is two phases: **initial state tagger** and **learning phase**. In the process of TBL, the TBL take untagged corpus as input and initial tagger tag likely tag for the tokens in the untagged corpus, then result temporary corpus as output. Then the learning phase take two corpus which is temporary corpus and reference or goal corpus which is manually tagged corpus assumed as correctly tagged corpus, and compared temporary corpus with goal corpus for rule derivation. The temporary corpus passes through the learning phase iteratively to derive rule transformation. The learning phase learns continues until no change of rule to improve temporary corpus compared with goal corpus. In the process of this, the learning phase produce ordered list of rules which can be applied and tagged untagged text. Below the figure 4 shows the design of transformation based learning.

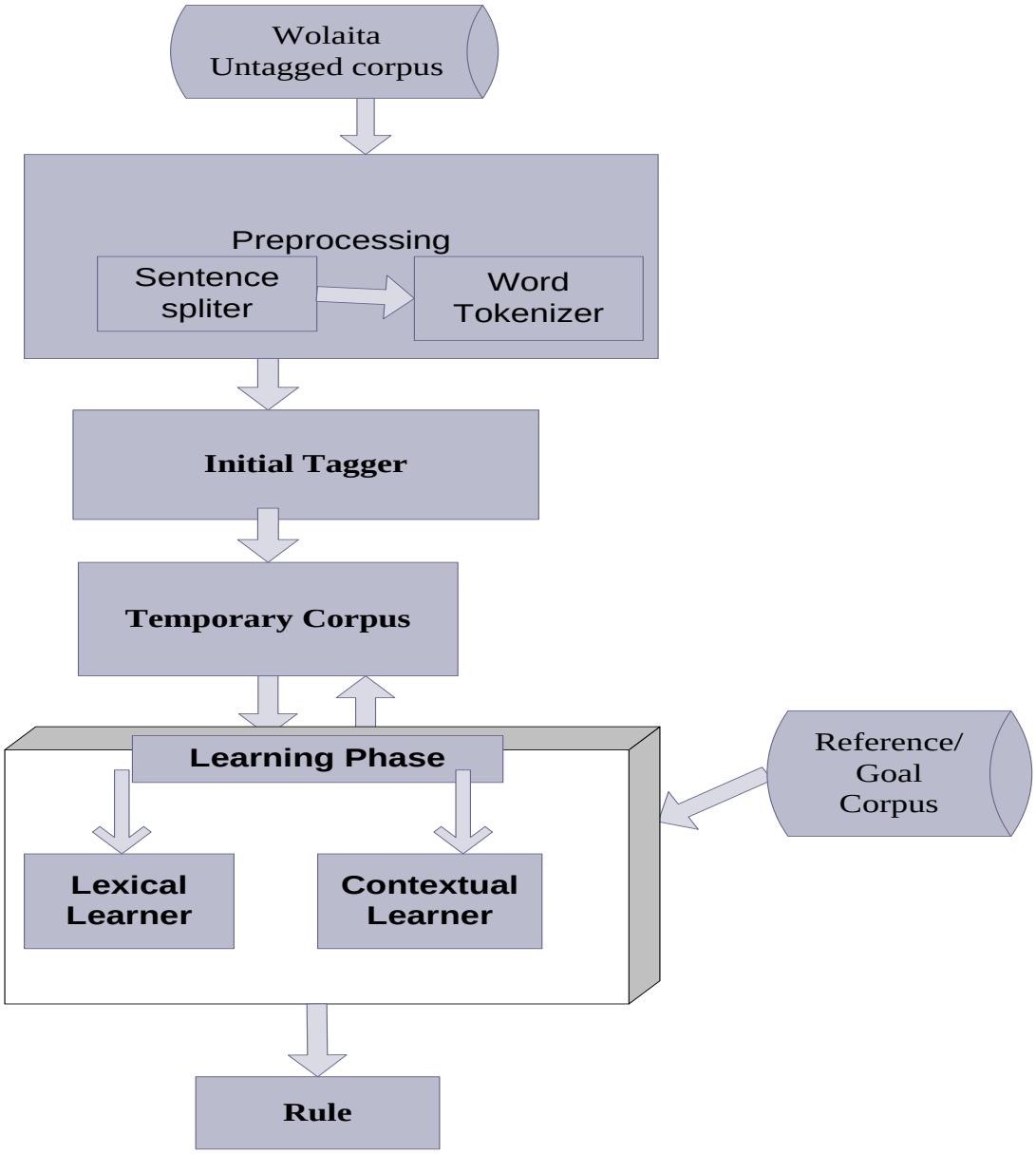


Figure 4: Transformation Based Learning Tagger Proceess

It can be seen from the figure above that, there are two learner phases, lexical learner and contextual learner. The lexical learner module uses a list of words containing information about the frequencies of tags, which is derived from the reference corpus but the temporary corpus for the lexical learner is a list of words which is similar to the reference corpus but tagged in

different way. Instead, the reference corpus for the contextual learner is the manually tagged corpus which is already running and the temporary corpus is the same corpus which was tagged in different way than the reference corpus.

The rule part contains two main part which is triggers (condition or current tag) and resulting tag. These rules are initialized from already defined transformation templates. Initially, these templates are defined as uninitialized variables which can later be instantiated to some rule throughout the training. Following is the form of these template rules:

i. If Trigger, then change the tag X to the tag Y
ii. If Trigger, then change the tag to the tag Y where X and Y are variables.

In template (i) is change the current tag X of a word to the resulting tag Y if

the rule triggers the word while template (ii) change the word tag regardless of its current tag to the resulting tag Y when the rule triggers the word. Based on this, a set of all rules is generated from the existing predefined templates.

i. Initial Tagger

The initial tagger takes untagged corpus as input and tag the untagged corpus based on statistical model which is N-gram model like unigram, bigram, trigram or default tagger. In this thesis, different tagger as initial taggers used which are unigram bigram and trigram taggers compared with the performance of Brill tagger. The initial tagger select depending on the final performance of the TBL tagger.

a. Unigram tagger:

Unigram is a type of the simplest n-gram (with$n$=1), which is a probabilistic language model for estimating next word in a sequence of n words. Without considering preceding word, Unigram calculates the probability of the next word and also assigns the probabilities to entire sequences. The probabilities can be estimated from a training corpus. When Unigram encounters a word that does not exist in the training corpus, Unigram assigns it a default tags 'None'[53]. For calculating the unigram probability, first determined that the frequency of each word occurs in the corpus. Probability of tag with given word is the frequency of tag with give word divided by the total number of word in training corpus.

P (ti/wi) = freq (wi/ti)/freq (wi)…….(4.9)

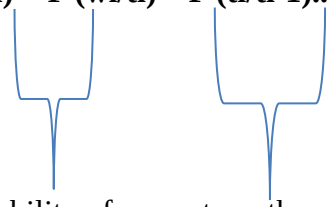
The above question (4.9) shows that how unigram tagger compute probabilities of tag give word. Here Probability of tag given word is computed by frequency count of word given tag divided by frequency count of that specific word. Corresponding probabilities will be checked after calculating frequency. Finally on the source of those probabilities final tagged output will be generated.

b. Bigram tagger

It applied statistically the same process as the unigram tagger with one specific difference. Different of bigram tagger from unigram tagger is it considers the context when assigning a tag to the current word. The bigram tagger is assign tag give word based on the preceding and current tags and it assumes that probability of tags depend on previous tag.

The basic idea behind all the statistical method is to capture most likely tag sequences for text, Thus this phenomenon can show by equation(4.10):[54].

P (ti/wi) = P (wi/ti) * P (ti/ti-1)..... (4.10)



The probability of current Word given current tag

the probability of a current tag given the previous tag

These probabilities are computed by equation (4.11)

P (ti/ti-1) =f (ti-1, ti)/f (ti-1)..... (4.11)

c. Trigram Tagger

It is second order Markova model that consider triples of consecutive words that means it checks for the occurrences of the words together with two words before.

ii. Learning Phase

Transformation-based learning is learning rules from the correctly prepared manually tagged corpus. In learning phase transformation based learning approach has two major sub-components that are used for learning rules from manually prepared corpus. Those sub-components are: the lexical learning rule and the contextual learning rule. The two sub-components are discussed as follows

a. Lexical Learning Rule

The aim of lexical rule learner is used to drive set of acceptable lexicon rule that can produce assign the most likely tag for any word in the given input text that may or may not be seen in the training tagged corpus. The lexicon is computed using a statistical method and it holds every word within the training corpus associated with its most frequent tag. It is used to tag untagged words that are seen at list one time during the training stage. The lexical rules generate and the lexical rule learner accept untagged Wolaita text and passes to the initial tagger, then the initial tagger to produce a temporary corpus termed TC0. After this, it finds the rule which becomes the best permissible score when applied to TC0 based on the condition which is predefined lexical rule templates.

Based on the result of the rule on the word to be tagged, the score of the rule may become positive, negative or zero. A best score for a rule means that a rule that gives better similarity with the gold or manually tagged text when applied to TC0. The score of rule is positive (+) means the rule changes the tag of the word from incorrect to correct. Negative (-) score of rule means the rule change the tag of the word from correct to incorrect and zero (0) means condition of the rule is not fulfilled. Applied to the first temporary corpus (TC0) after computing rule with best score, then produce the next temporary corpus called (TC1) and added best score rule to the set of rules. The procedures continue in the same way to produce the entire permissible rule with the corresponding temporary text until no rule can change the tag of temporary corpus.

Templates that are used in lexical rule learner component are given below

1. Change the most likely tag to Y if the current word has suffix X
2. Change the most likely tag to Y if adding/deleting the suffix X, |X|<9, results in a word, |X| is length of x.
3. Change the most likely tag from X to Y if deleting/adding the suffix (character) X, |X|<9, results in a word, |X| is length of x.
4. Change the most likely tag from X to Y if word W ever appears immediately to the left/right of the word.

The above rules template are prepared for Wolaita language In the case of Wolaita language, the numbers of suffix are extended up to 9 characters based on the corpus prepared for this research. The lexical template prepared considering the feature of Wolaita language.

For instance:

Ammanekketanne --→Amman-ekkettanne

Ixxidaakkonne->Ixxi-daakkonne

So, extended the suffix to 9 characters for Wolaita language and it changes the original brill tagger template of lexical learner rule number 2 and 3. The lexical rules formed by the lexical learning module are then used to initially tag the unknown words in the contextual training corpus. Some of the lexical rules extracted by lexical modules from training corpus are presented in appendix **B**.

b. Contextual rule

Contextual rule learner is used to tag unknown words based on context or environment. It is learned for disambiguation and better accuracy using contextual rule learner, the learner takes initial temporary and gold corpus as an input then, the learner generates possible rules for predefined contextual rule template when the condition is satisfied. The goal of contextual rule learner is the same way of that of lexical rule learner, they generate set of all permissible rule. But the permissible rule of contextual rule learner different from permissible rule of lexical rule learner because the two modules uses different predefined transformation templates and the lexical rule learner build rule based on morphology of word, the contextual learner depend context of the that words. Following are some templates of the contextual rule learner.

1. The preceding/following word is tagged with X
2. One of the two preceding/following words is tagged with X
3. One of the three preceding/following word is tagged with X
4. The preceding word is tagged with X and the following word is tagged with Z
5. The preceding/following two words are tagged with X and Z
6. The two words before/after is tagged with X
7. The current word is X and the following word is Z.
8. The current word is X and the preceding/following word is tagged with Z.

9. The current word is X and the word two words before/after is tagged with Z.

Some of the contextual rules extracted by contextual modules from training corpus are presented in appendix **C**.

The transformation based learning sub-components generated all possible rules, after that the learner computes the score of each rule for a specific word. The learner picks rules with highest score and stores it in sub-component of transformation based learning tagger which is rules based on the score.

Scores on the temporary corpus TC0 are computed for all rules in permissible score if the trigger condition is satisfied for the rules. Score of rule R can be computed simply by comparing the word W tag in CT1 when rule R1 is applied with the correct tag of same word in the reference corpus. If the applied rule R1 corrects the tag of the word, the score of R1 is +1. Though, if the applied rule R1 introduces error, the score for R1 is -1. In all other cases, the score for the rule is 0. Consequently, the score for rule R is generalized as:

R= number of errors corrected - number of errors introduced

It picks the rule with the highest score and then put on output list. Hereafter, the learner applies R1 to TC0 and produces TC1, on which the learning continues. The process continues in the same manner until no rule occurs that makes the temporary corpus look like with that of the reference text.

The Learning Algorithm

Transformational rule based learning algorithm for both of the contextual and lexical rule can be generalized as follow:

1. Initial tagger takes untagged sentence as input then tag
2. Learning lexical and contextual rules from initial tagger by repeatedly calculating the error scores then:
    i. Pick the highest score of rule and add it to rule set
    ii. Use rule set to tag the initialized text
    iii. Repeat step 2 until no rule to change the state of the tag.
    iv. Lastly the output of lexical and contextual rules.

4.2.4.3 Hybrid Tagger

In this research, the implemented hybrid tagger is two-step process. The first step process is performed by HMM tagger and second step process is performed by rule based tagger. The HMM tagger first take untagged text and assigns tag to raw text based on probability and proved optimal level of tag sequence. The threshold value is not attained in HMM tagger, it is corrected by second step process which is rule based tagger. Rule based tagger correct the error of tag sequence by applying transformation rule that it has learned during training time. The threshold value is fixed based on the result of performances of hybrid tagger in experiments.

The HMM tagger use Viterbi algorithm to find the optimal probabilities of the tag/word pairs and the probabilities of optimal path. Thus, it is possible to compare the probabilities of each word tag is greater than the fixed threshold value. When the probability of the assigned tag by HMM tagger for the given word is greater than the fixed threshold value it ensure that the assigned tag by HMM tagger is correct tag or does not need correction by rule based tagger. Otherwise, the word is given to rule based tagger to correct tag. hence, the proposed hybrid tagger accept untagged text and tag using HMM tagger which acts as an initial tagger for the raw text to be tagged and the rule based tagger correct the output of HMM tagger by applying rules if the predetermined threshold value is not attain.

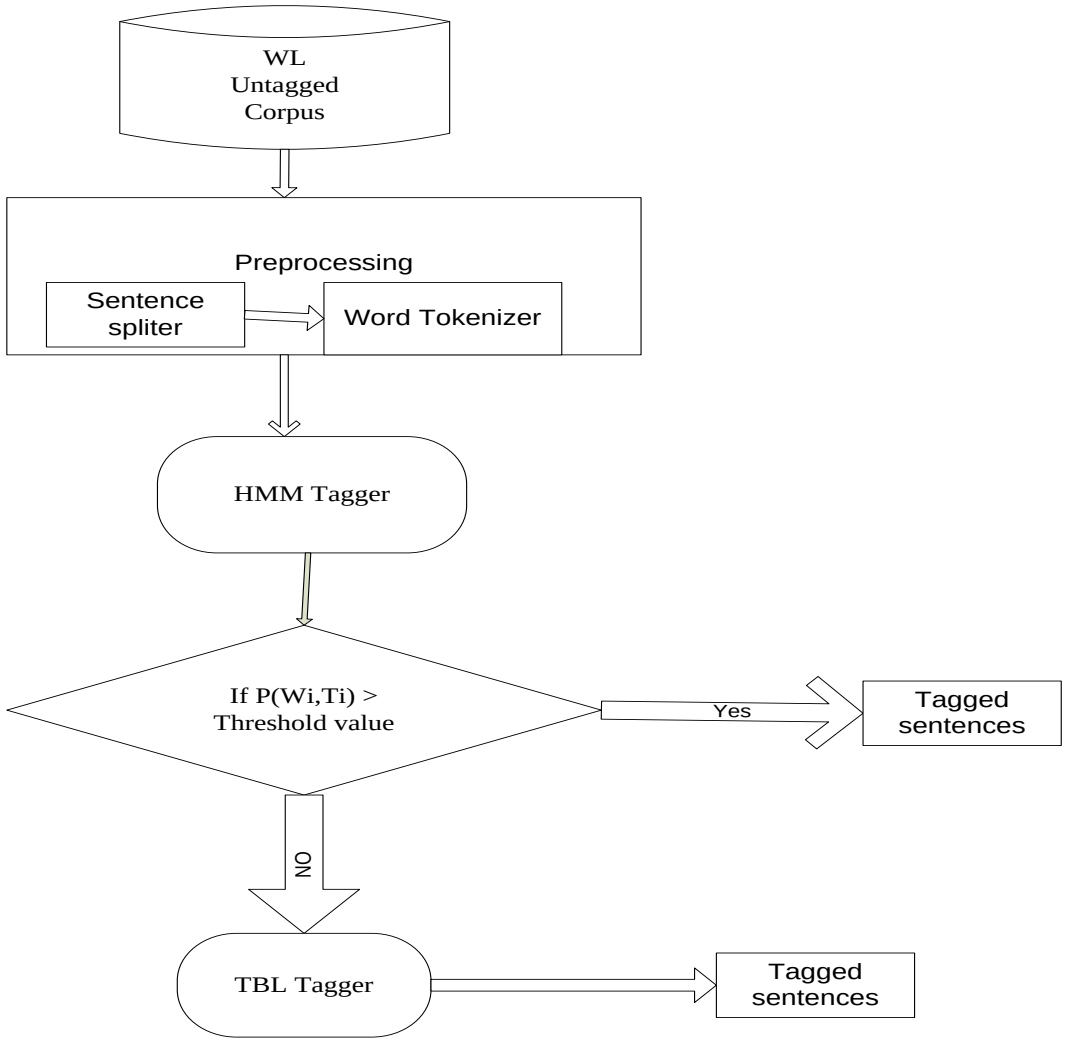


Figure 5: Design of Hybrid Tagger

The untagged texts of Wolaita language are given to the preprocessing component. Then the word sequence Wi is given to the HMM tagger as an input; the HMM tagger assign the tag sequence Ti using Viterbi algorithm (which find the best optimal path) and the output of the HMM tagger is the word tag sequence (Wi,Ti). This word tag sequence is given to the output analyzer that checks whether the determined threshold value for a word Wi is achieved or not. The threshold value is a value used for checking the sureness level of tagging a given sequence of words. So, the output analyzer decides based on the threshold value. Consequently if the threshold value of a word tag pair is lower than the fixed threshold value, a fixed window size, in this case a window size of two which implies bigram of words is given to the transformation

based learning tagger for correction and the transformation based learning tagger produce the corrected tagged words. Otherwise the HMM tagger output is the final output of the word to be tagged. This process is repeated until the HMM tagger tagged words are checked by threshold value and corrected by transformation based learning tagger

4.2.5. Test and evaluation methods

The researcher used percentage split method to test and evaluate the model. To measure the accuracy, the entire corpus was divided into two dataset: training data set and test data set. The gold standard tags are used to compare and estimate percentage accuracy of the model. The model tags test dataset sentences based on the trained knowledge and then evaluate model by count correct tags from test data set and divided by the total number of test data set (incorrect and correct tags).

$$\% \text{ accuracy} = \frac{\text{Number of correct tags}}{\text{Number of incorrect tags + number of correct tags}} \text{ ---------- (4.12)}$$

4.3. Materials used

In the proposed research, the following software tools were used.

- ✓ **Python**

The Python programming language is object-oriented interpreted language, a dynamically-typed, Although, its primary strength lies in the ease with which it allows a programmer to rapidly prototype a project, its powerful and mature set of standard libraries make it a great fit for large-scale production-level software engineering projects as well. Python is commonly used in Artificial Intelligence (AI), Natural Language Generation(NLG), Neural Networks and other advanced fields of Computer Science[55]. Python is a simple but powerful programming language with excellent functionality for processing linguistic data[56]. It has efficient and high level data structure with simple but effective approach to object-oriented programming [27].

- ✓ **Microsoft Visio 2013 software-**to draw the figures.
- ✓ **NLTK**

NLTK, the Natural Language Toolkit, is a suite of open source program modules, tutorials and problem sets, providing ready-to-use computational linguistics courseware. NLTK

covers symbolic and statistical natural language processing, and is interfaced to annotated corpora[57]. It is a leading platform for building Python programs to work with human language data. The Natural Language Toolkit (NLTK) is a platform used for building Python programs that work with human language data for applying in statistical natural language processing (NLP).

The Natural Language Toolkit is an open source library for the Python programming language originally written by Steven Bird, Edward Loper and Ewan Klein for use in development and education. It contains text processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning. It also includes graphical demonstrations and sample data sets as well as accompanied by a cook book and a book which explains the principles behind the underlying language processing tasks that NLTK supports[58].

CHAPTER FIVE

5. RESULT and DISCUSSION

5.1. INTRODUCTION

This chapter describes the experimental results and respective discussion for application of hybrid approach for Wolaita language part of speech tagger. In order to implement this study the researchers have used python programming language packages. To develop the HMM model, Transformation based learning and Hybrid (HMM & Transformation based learning) model for Wolaita PoS tagger. After comparing the two models performance, the author used HMM as initial tagger and TBL as a corrector tagger.

5.2 Experiment Results

In this research, the author evaluated three experiments with similar validation method and with the same corpus. These are HMM tagger, TBL and hybrid tagger. In hybrid tagger there are five experiments by using different threshold value. The researcher used percentage split evaluation method in each individual experiment. According to this percentage split evaluation method, the entire corpus is split into two sets: Training set and testing set. The entire corpus used for this research is 1134 sentences or 14,358 words. To decide the splitting size of training and testing dataset the following experiments were done by using different percentage split.

Table 9: Experiment Results of HMM Tagger using different Train /Test Split

	70/30(70% training set and 30% test set)	80/20(80% training set and 20% test set)	90/10(90% training set and 10 test set)
HMM Tagger Accuracy	72.52%	73.91%	88.14%

In HMM tagger there is three experiments to decide the percentage of training set data and test data. The researcher used three percentage split: those are 70/30, 80/20 and 90/10. The accuracy of HMM tagger is 72.52%, 73.91% and 88.14% of 70/30,80/20 and 90/10 respectively. From those 90/10 split accuracy of tagger is higher. So, the researcher decides 90/10 percentage split

based on the experiment result. Table 10 shows experiment result of TBL tagger for each percentage split.

Table 10: Experiment Results of TBL Tagger using different Percentage Split

Initial Tagger	70/30(70% TDS and 30% TS)	80/20(80% TDS and 20% TS)	90/10(90% TDS and 10% TS)
Unigram Tagger	76.61%	78.04%	88.66%
Bigram Tagger	75.60%	77.81%	90.69%
Trigram Tagger	76.16%	78.18%	92.96%

The above table 10 showed that the experiment result of TBL tagger of three different percentage split and the percentage split of 90/10 (90% training data set and 10% test data set) is outperformed others. So, the researcher decided 90/10 percentage split for this study based on the experiment result.

5.2.1. Experiment Result for HMM Tagger

In order to test the performance of the HMM tagger like that of transformation based learning tagger. From the entire corpus, 90% (1020 sentences) is used for training and the remaining 10%(114 sentences) is used for testing purpose. To conduct the experiments, first the entire training dataset divided into ten parts, started training HMM tagger the first 10% (102) sentences of training dataset. After the HMM tagger is trained using 10% of training dataset, performance of trained tagger is measured by test dataset and repeated this process by adding training dataset by 10% until 100% training dataset.

Table 11: Shows different Experiments with different Portion of Training-dataset for HMM Tagger

Training-set	10% (102)	20% (204)	30 (306)	40% (408)	50% (510)	60% (612)	70% (714)	80% (816)	90% (918)	100% (1020)
HMM-tagger-accuracy	76.45%	77.61%	77.32%	77.26%	78.08%	78.43%	78.37%	78.89%	79.01%	88.14%

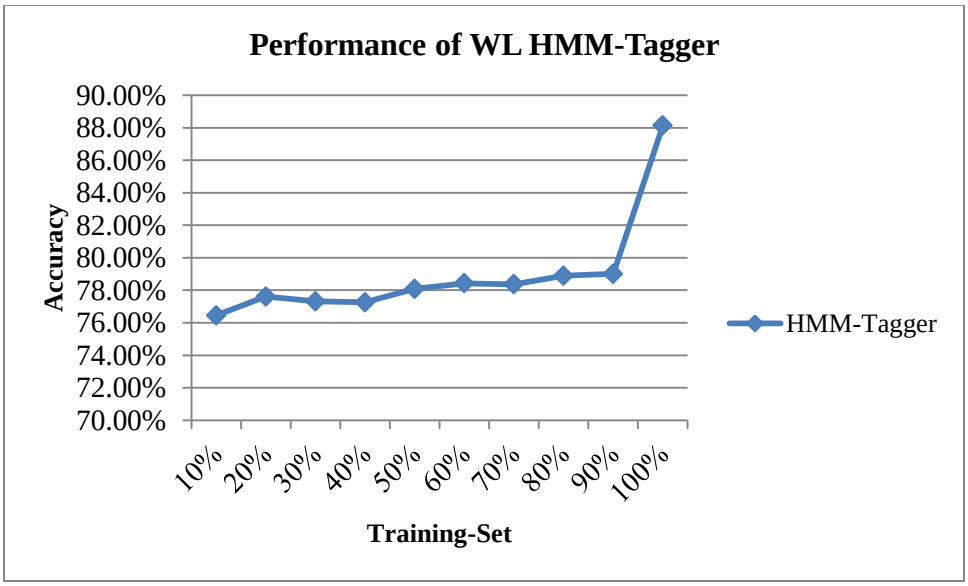


Figure 6: Performances of HMM Tagger (Viterbi)

The above Figure 6 shows the performance of HMM tagger for Wolaita language when the system takes different amount of training dataset incrementally and the same amount of test dataset. Also it describes that the characteristics of HMM tagger, because the HMM tagger require a large amount of training dataset to give better performance. In this case, when training dataset is 100% the accuracy is 88.14%.

5.2.2. Experiment Result for Transformation based learning

In transformation based learning, first the researcher create an un-annotated version of the training corpus and then pass it to the initial state annotator (unigram, bigram and trigram). The researcher used initial state annotator, which is simple statistical techniques/n-gram tagger (unigram, bigram, trigram and Regexp). Firstly the un-annotated data is passed to unigram tagger; the unigram tagger by using the maximum probability P (ti/wi), finds the best tag and label it to the targeted word.

t= argmax P(t/w) this computed as P(t/w)= C(t,w)/C(w), where C(w) is number of time the word w appear and C(t,w) is the number of time tag t appear with word w.

In the case of unigram tagger, assigns a 'None' tag to all unknown tokens. The researcher used regexp tagger as backoff for handling unknown words as it gives better score than the default tagger. Now the regexp tagger is used as a backoff, thus that when unigram tagger fails to

determine a tag, it refers backoff. Consequently, it employed as backoff tagger, which uses simplest word suffixes for determining part of speech tag. For instance, some suffix of Wolaita language word used for unknown words to accurate unigram tagger. Regexp tagger is backoff tagger for unigram tagger but not initial state annotator by itself on this research.

```
r'^-?[0-9]+(.[0-9]+)?$', 'AJN'), (r'.*(ntta|ntto)$', 'ON'),
(r'^-?(Ha|ha|He|he|hegaa|Hegaa|Hagaa|hagaa)?$', 'DD'),
(r'.*(sonaa|tiis|taasu|iis)$', 'VV'), (r'.*di$', 'VC'),
```

After un-annotated data annotated by unigram tagger is passed to learner; the data tagged by initial tagger (unigram tagger) is temporary corpus. This temporary corpus then compared with goal corpus/reference corpus to check the correctness of the initial annotator. Temporary corpus then turn into input to the learner. The learner is learned an ordered list of transformation, which is applied on temporary corpus to make it near to the goal corpus or reference corpus. Transformation is contained two components: (rewrite rule and triggering environment). For instance of rewrite rule and triggering rule are as follows:

Change the tag from NN to NC → rewrite rule
The preceding word is PUN → triggering rule

The rules are basically instantiated from a set of predefined lexical and contextual transformation templates. Predefined template comprised un-instantiated variables.

If trigger, then change the tag x to the tag y, where x and y are variable

The clarification of the transformation template is the rule triggers on a word with current tag x then the rule replaces current tag with resulting tag y. In training time, tagger estimates the values for x and y and the variable stated in the context, to create thousands of candidate rules. In each iteration of learning pick the rule and applied on the temporary corpus provides best score according to its value of;

Score = Fixed – Broken, where Fixed = number of tag changed incorrect ->correct
Broken =number of tag changed correct -> incorrect

Subsequently, adding the best scored rules in the final ordered list of rule, the whole corpus is update by applying this high scored rule. In this manner, learning continues until there is no

change for further improvement. Predefine transformation template divided into two which are lexical and contextual transformation template.

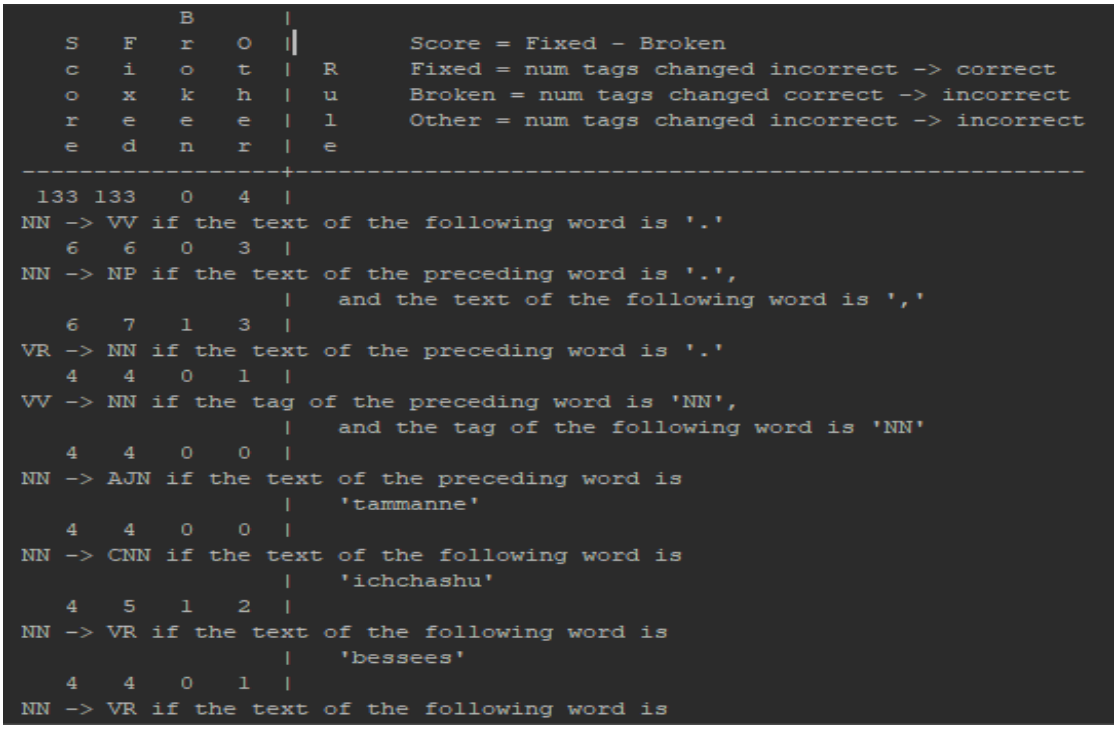


Figure 7: Sample Screenshot Transformation Template Generated from Training Corpus

The above figure 7 represents the some transformation template generated from training corpus based on predefined transformation template. The list of rule listed in decrease order based on score. For instance, the first transformation in figure 7 NN->VV the number of score is 133, that mean incorrectly tagged 133 NN tag to correct tag to VV based on the following word of (.). Score = fixed –broken------→ 133-0 =133

VR-> NN if the text of the preceding word is (.). In this case incorrectly tagged tag VR to correct tag NN based on the preceding word of (.) and one correctly tagged tag VR to incorrectly tagged tag NN and three incorrectly tagged tags to incorrectly tagged tag. Score = 7-1=6

After being annotated by the unigram tagger, brill algorithm learned by taken temporary corpus from unigram tagger (initial tagger) and reference corpus, after learned to generate lexical and contextual rule to tag untagged sentences based on the learned rule. The modeled tagger accuracy was computed by test dataset. Second the researcher was used bigram tagger as initial

tagger for brill tagger development. Bigram tagger is similar to the unigram tagger but it uses both current tag for current word and preceding tag to find the most likely tag for each word. The brill algorithm taken temporary corpus from initial tagger (bigram tagger) and leaned the lexical and contextual rule from training dataset by comparing temporary corpus with reference corpus. Then after that, the accuracy of brill tagger was computed by test set. Lastly, the researcher used trigram tagger as initial tagger and computed the accuracy of brill tagger. The trigram tagger similar with that of bigram tagger but the only difference is, it uses three tags current tag for current word and two preceding tag. As result the performance of TBL tagger computed using three split method. As result, the performance of TBL tagger computed using three initial taggers with different portion of training sets. The following table 12 shows the experiment result of transformation based learner/rule based tagger. The performance of rule based tagger tested using ten different experiments with different portion of training dataset by three different initial taggers namely unigram, bigram and trigram tagger.

Table 12: Shows different Experiments with different Portion of Training dataset and Initial Taggers for Rule-based Tagger

Training set	10% (10)	20% (204)	30% (306)	40% (408)	50% (510)	60% (612)	70% (714)	80% (816)	90% (918)	100% (1020)
Unigram Tagger	87.96 %	87.50 %	88.08 %	87.85 %	88.19 %	87.96 %	88.14 %	88.14 %	88.31 %	88.48 %
Bigram tagger	90.40 %	90.40 %	90.52 %	90.40 %	90.34 %	90.58 %	90.46 %	90.34 %	90.58 %	90.69 %
Trigram tagger	92.55 %	92.55 %	92.67 %	92.61 %	92.73 %	92.84 %	92.73 %	92.61 %	92.79 %	92.96 %

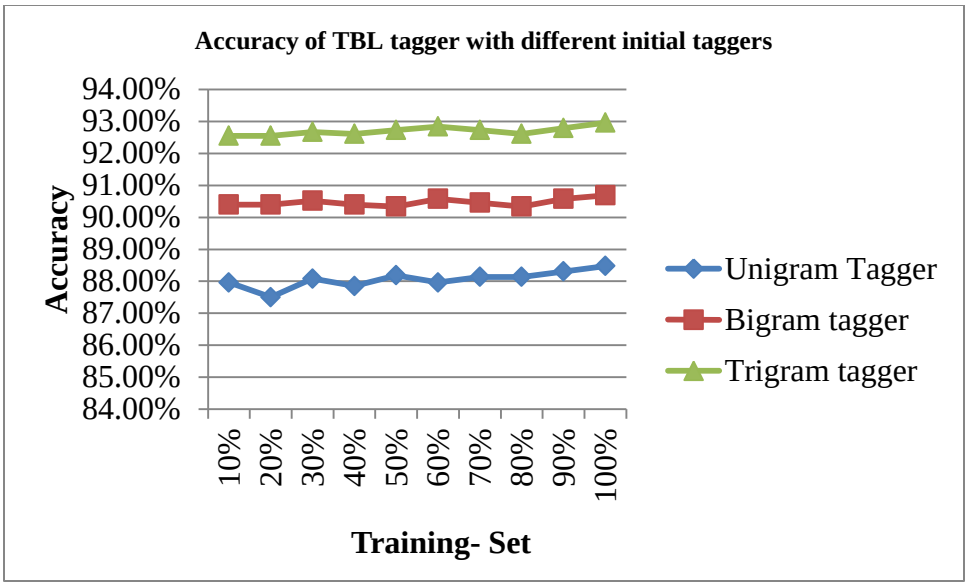


Figure 8: Rule Based Tagger Performance for Wolaita Language

The above figure shows different experiment results of rule based tagger. In this thesis work, three initial taggers (unigram, bigram and trigram tagger) used with different portion of training dataset. Started training tagger first by 10% of training set and 10% of test dataset to test the accuracy of brill tagger and continue testing by add 10% training dataset until 100% training set. When the training dataset increase the performance of the taggers increase and the performance of rule based tagger used trigram taggers as initial tagger greater than bigram and unigram tagger. The performance of rule based tagger is 88.48%, 90.69% and 92.96% of unigram, bigram and trigram respectively.

5.2.3. Experiment Result for Hybrid Tagger

The designed hybrid taggers for Wolaita language are combined HMM and rule-based tagger. In this hybrid model, the HMM tagger first annotate the word sequence within the sentence and if the desired threshold value of the given sentence is not attained, the sequence of word is given to the rule based tagger for correction. In this research, the HMM tagger is used as initial annotator and rule-based tagger as a corrector. The fixed threshold value for this study is 0.6 since taking threshold value less than 0.6 does not bring significant difference on the performance of the tagger which means less than 0.6 give to TBL tagger for correction. Rule based tagger corrects more words when the threshold value increase. As a

result of this, the hybrid tagger gives better performance when the threshold value goes up, As it is presented in table 13. Hence, the fixed threshold value to 0.6, performance of 94.82% is obtained. But the threshold value increase to 1, the performance of tagger less accurate because the probability always between zero and one. Table 13 indicates the performance of hybrid tagger with different threshold value.

Table 13: Performance of hybrid Tagger

Threshold-Value	0.1	0.2	0.4	0.5	0.6	0.8	1
Performances	90.696	92.012	93.932	94.155	94.826	94.822	90.400

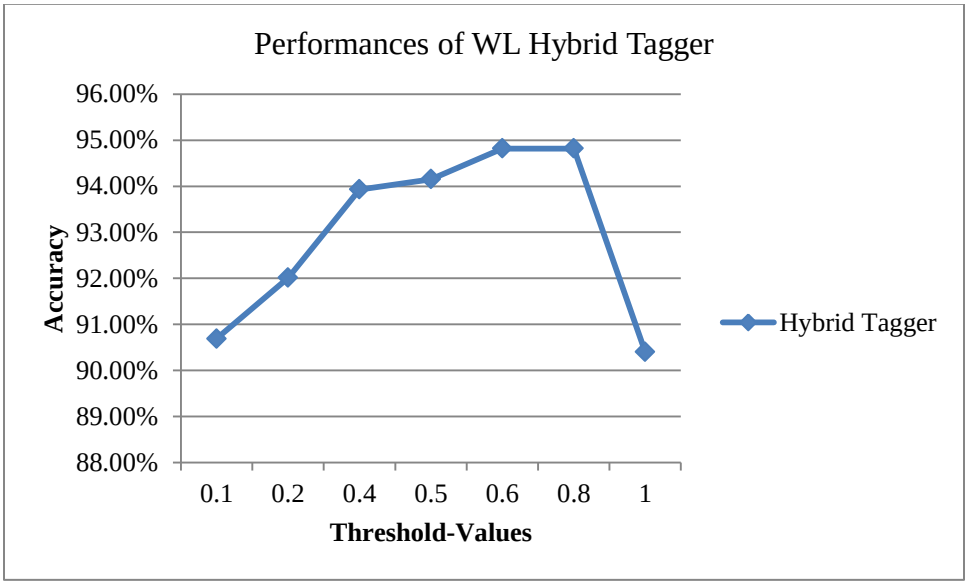


Figure 9: Performance of Hybrid Tagger

The above figure 9 shows the performance of hybrid tagger that consists HMM and TBL tagger. In the graph the researcher used different threshold value between 0 and 1. The performance result of hybrid tagger in threshold value 0.6 and 0.8 is 94.826, 94.822 respectively. The fixed threshold value is 0.6 based on the performance result because the performance result of threshold value 0.8 is relatively less accurate than the threshold value of 0.6. The threshold value increase, transformation based learning tagger corrects more words. As a result of this, the hybrid tagger gives better performance as the threshold values increase. But when the threshold value closed to 1.0 the accuracy goes down because most of

the time the probability is not greater than or equal to 1, in this case TBL tagger tags all the words. So, the fixed threshold value 0.6, overall performance of hybrid tagger 94.82% is achieved.

Hence, this study is different from previous work of Berhanu H. because the researcher developed PoS tagging for Wolaita language using small corpus and HMM and CRF approach used individually. For annotation of corpus he used 22 tagset; however, the final output produced less accuracy because of the researcher has used small data. But HMM approach requires large amount of data for better performance. Thus, in this research study the hybrid approach used with relatively large corpus in order to improve the accuracy of tagger. So, this research work is different in terms of approaches, tagsets and corpus from previous work of Berhanu H.

5.3. Performance analysis

The performance analysis of HMM, rule-based and hybrid tagger of Wolaita language in relation with different part of speech tag, entire corpus frequency of tags, training dataset and testing dataset are considered. In this thesis work, 26 tagset was identified and used to train the model. Hence, this entire tagset divided into two parts: most frequent tags in Wolaita language and remaining tag is derived tags which are represented by others tags. For performance analysis eleven basic tagset of Wolaita language analysis independently and 15 derived tagset are categorized as others tagset like NV,PPRP,ADVC,ADJC,DPRP,PC,NP, CNN,VC,ON,…

The frequency of entire corpus, training dataset and test dataset of tagset is presented using tables as follows:

Table 14: Frequency of Entire Corpus, Training-dataset and Test-dataset

Tag	Tag frequency in entire corpus	Tag frequency of training corpus	Tag frequency of testing dataset
NN	3413	3039	367
PUN	1822	1542	276
VV	1249	1086	146
ADJ	856	771	78
ADV	564	493	45
PP	622	438	103
PRP	446	379	54
DD	320	287	33
CJ	291	268	23
AJN	221	211	10
INT	15	11	2
OTHER	4539	4158	538
TOTAL	14,358	12,683	1,675

CHAPER SIX

6. CONCLUSION AND RECOMMENDATION

6.1 Conclusions

Part of speech tagger is one of natural language processing application. PoS tagging is the essential basis for other Natural Language Processing (NLP) applications and it's a research area in the field of natural language processing. It is the process of assigning corresponding part of speech tag for a word in sentence. The most common approaches to develop part of speech tagger identified by different researchers are: rule based, HMM, artificial neural network, memory based and hybrid of individual approaches. For this thesis, a hybrid approach used that combines HMM and TBL tagger at sentence level is designed for Wolaita language. This paper describes a sequence of different PoS tagging experiments and to develop this hybrid model the HMM and TBL tagger were developed and evaluated three tagger individually those are rule based tagger, HMM tagger and hybrid tagger.

For the purpose PoS tagger development manually tagged corpus used, which consist of 1134 sentences (14,358 words) and 26 tagsets are prepared for corpus annotation. Because in the language there is no standard corpus and tagsets for natural language processing. So, corpus and tagsets are manually prepared for the study. The tagged entire corpus is divided into training and testing dataset. The researcher decided to use 90/10(90% training dataset and 10% testing dataset) of entire corpus based on experiments of table 9 and 10. The 90% of the entire corpus for training and the remaining 10% of corpus for testing.

NLTK and Python version 3.5.0 are used in the implementation and experiment of Wolaita language part-of-speech tagger. To evaluated the performance of three types of tagger namely HMM tagger, rule-based and hybrid tagger with the same training dataset and testing dataset conducted by different experiments. Consequently, 88.14%, 92.96% and 94.82% performances are obtained for HMM, rule-based with trigram initial state tagger and hybrid taggers respectively.

The performance of hybrid tagger was tested on different threshold values and threshold value of 0.6 scored better performance than other threshold values. So, 0.6 was used as fixed

threshold values of Wolaita language hybrid tagger. When compare the accuracy of hybrid tagger with HMM and rule-based tagger individually, hybrid tagger increased by 6.68 than HMM tagger and 1.86% than rule-based tagger.

Among these three tagger the hybrid approach outperformed the rule based and HMM tagger. Thus, the hybrid approach is better for classifying tag for word of Wolaita language at the sentence level. Accordingly, based on their performance, the study has concluded that the hybrid tagger performs better than two approaches HMM and rule-based tagger taken separately.

6.2 Recommendation

Part of speech tagging is pre-processing component for many advanced or high level NLP applications such as parsing, word-sense-disambiguation, machine translation, text-summarization, question –answering etc. Therefore, the researchers who are interested to do research in the area of NLP application for local language and part of speech tagging of Wolaita language in particular that should be recommended for future research in the area of natural language processing. These recommendations are listed below.

- ✓ Comparison this work with other hybrid approaches like: rule based and Artificial Neural Network or rule based and conditional random field they may found different result.
- ✓ Comparative study of HMM based, rule based, conditional random field and Artificial Neural Network based taggers for Wolaita language.
- ✓ Extending this work with large training data and identified tagset using different features of a language not included in this thesis: gender, tense etc.

Reference

[1] Getachew Mamo, “Part-of-Speech Tagging for Afaan Oromo Language,” Addis Ababa Univerity Msc Thesis, 2009.

[2] J. Perkins, *Python 3 Text Processing with NLTK 3 Cookbook*. PACKT Publishing Ltd., 2014.

[3] G. Y. Bade and H. Seid, “Development of Longest-Match Based Stemmer for Texts of Wolaita Language,” vol. 4, no. 3, pp. 79–83, 2018.

[4] Tewodros Abebe Gebreselassie, “Text-To Speech_Synthesizer_For_Wolaytta,” Addis Ababa University,Ethiopia, 2009.

[5] H. Fanta, “Speaker Dependent Speech Recognition For Wolaita Language,” Addis Abeba University, Msc Thesis, 2010.

[6] L. Lessa, “Development of Stemming Algorithm for Wolaita Language,” Addis Ababa University, Msc Thesis, 2003.

[7] M. Getachew, “Automatic part-of-speech tagging for Amharic language an experiment using stochastic Hidden Markov Approach,” Addis Ababa University, Msc Theesis, 2005.

[8] Y. Biadgo, “Application of multilayer perception neural network for tagging part-of-speech for Amharic language,” Addis Ababa University, Msc Thesis.

[9] H. Berhanu, “Part Of Speech Tagging for Wolaita Language,” Addis Ababa University, Msc Thesis, 2015.

[10] M. Wakasa, “A Descriptive Study of the Modern Wolaytta Language,” Doctoral Dissertation, University of Tokyo., 2008.

[11] Z. Mekuria, “Design and Development of Part-of-speech Tagger for Kafi-noonoo Language,” Addis Ababa University, ,Msc Thesis, 2013.

[12] G. Belay and M. Zeleke, “Designing a Stemmer for Kafi-noonoo Language,” *IJESC*, vol. 9, no. 10, pp. 1–4, 2019.

[13] G. Emiru, “Development of Part of Speech Tagger using Hybrid Approach,” Addis Abeba university, Msc Thesis, 2016.

[14] A. L. and M. M. Meryeme Hadni, Said Alaoui Ouatik, “Hybrid Part of Speech Tagger for Non-Vocalized Arabic Text,” *Int. J. Nat. Lang. Comput.*, vol. Vol.2, 2013.

[15] L. Synopsis, “Natural Language Processing.” pp. 2003–2004, 2004.

[16] D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural Language Processing: State of The Art, Current Trends and Challenges,” 2017.

[17] T. G. Abreha, “Part of Speech Tagging for tigirgna Language,” Addis Ababa University,Msc Thesis, 2010.

[18] M. M. Z. E. Mohammed, E. Bashier, and M. Bashier, *Machine Learning: Algorithms and Applications*, no. December. London New York: ResearchGate, 2016.

[19] A. Dey and A. S. Learning, “Machine Learning Algorithms : A Review,” vol. 7, no. 3, pp. 1174–1179, 2016.

[20] L. Nandanwar and K. Mamulkar, “Supervised , Semi-Supervised and Unsupervised WSD Approaches : An Overview,” vol. 4, no. 2, pp. 2–6, 2015.

[21] S. Y. Hala and S. H. Virani, “Improve accuracy of Parts of Speech tagger for Gujarati language,” *Int. J. Adv. Eng. Res. Dev.*, vol. 2, no. 5, 2015.

[22] F. M. Hasan, “Comparison Of Different Pos Tagging Techniques For Some South Asian Languages,” BRAC,Dhaka,Bangladesh, 2006.

[23] B. Gopal, P. Khumbar, D. Dipankar, and D. Sivaji, “Part of Speech (POS) Tagger for Kokborok,” *proceding of COLING*, vol. 1, no. 1, pp. 923–932, 2012.

[24] M. Abubeker, “Part of Speech Tagger For Oromo Language Using Transformational Error Driven Learning(TEL) Approach,” Addis Ababa University, Msc Thesis, 2010.

[25] N. Adhvaryu and P. Balani, “Survey : Part-Of-Speech Tagging in NLP,” *Int. J. Res. Advent Technol. (E-ISSN 2321-9637)*, no. 1, pp. 102–107, 2015.

[26] D. Kumawat, "POS Tagging Approaches : A Comparison," *Int. J. Comput. Appl. (0975 – 8887)*, vol. 118, no. 6, pp. 32–38, 2015.

[27] Kamal Sarkar and Vivekananda Gayen, "A Trigram HMM-Based POS Tagger for Indian Languages," in *Proceeding of International Conference on Frontiers of Intelligent Computing*, 2013, p. pp 205–212.

[28] I. Daniel Jurafsky and James H.Martin. Prentice, *"Speech and Language Processing" An Introduction to Natural Language, Computational Linguistics and Speech Recognition*. United state of America: Alan Apt, 2000.

[29] K. Mohnot, N. Bansal, S. P. Singh, and A. Kumar, "Hybrid approach for Part of Speech Tagger for Hindi language," *Int. J. Comput. Technol. Electron. Eng.*, vol. 4, no. 1, 2015.

[30] K. G. Avinesh.PVS, "Part-Of-Speech Tagging and Chunking using Conditional Random Fields and Transformation Based Learning," in *Workshop on Shallow Parsing for South Asian Languages*, 2007, no. January.

[31] W. Black and J. Mcnaught, "Arabic Part-Of-Speech Tagging Using Transformation-Based Learning," *Manchester M1 9PL United Kingdom*, no. 2001, 2007.

[32] A. G. Ayana, "Towards Improving Brill's Tagger Lexical And Transformation Rule Afaan Oromo Language," Hawassa University, 2015.

[33] A. Kupusinac, "Transformation-based part-of-speech tagging for Serbian language," *Recent Adv. Comput. Intell. Syst. Cybern.*, 2011.

[34] M. Alex and L. Q. Zakaria, "Kadazan Part of Speech Tagging Using Transformation-based Approach," *Procedia Technol.*, vol. 11, no. Iceei, pp. 621–627, 2014.

[35] N. V.Patil, "International Journal of Computer Sciences and E pen A ccess POS Tagging for Marathi Language using Hidden Markov Model," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 1, pp. 409–412, 2018.

[36] M. Okhovvat and B. Minaei, "A Hidden Markov Model for Persian Part-of-Speech Tagging," vol. 3, pp. 977–981, 2011.

[37] A. G. Fatma Al Shamsi, “A Hidden Markov Model-Based POS Tagger for Arbic,” *Journees Int. d’Analyse Stat. des Donnees Textuelles*, 2006.

[38] M. Khanam, “Part-Of-Speech Tagging Of Urdu in Limited Resources Scenario,” *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 2, no. 10, pp. 3280–3285, 2014.

[39] K. Mohnot, N. Bansal, S. P. Singh, and A. Kumar, “Hybrid approach for Part of Speech Tagger for Hindi language,” vol. 4, no. 1, 2014.

[40] P. K. Dwivedi and P. K. Malakar, “Hybrid Approach Based POS Tagger for Hindi Language,” vol. 4840, no. August, pp. 63–68, 2015.

[41] K. N. R. N. Merin Francis, “Hybrid Part of Speech Tagger for Malayalam,” *IEEE*, 2014.

[42] A. R. W. Dilmi Gunasekara, W.V.Welgama, “Haybrid part of Speech Tagger for Sinhala Language,” in *International Conference on Advances in ICT for Emerging Regions(ICTer)*, 2016.

[43] P. Arulmozhi, “A Hybrid POS Tagger for a Relatively Free Word Order Language,” *ACADEMIA*.

[44] R. Simionescu, “Hybrid Part of Speech Tagger,” in *Proceedings of the Workshop “Language Resources and Tools with Industrial Applications,”* 2011.

[45] E. U. M. Fareena Naz, Waqas Anware, Usama Ijaz Bajwa, “Urdu Part of Speech Tagging Using Transformation Based Error Driven Learning,” *World Appl. Sci. J.*, vol. 16, 2012.

[46] M. Wakasa, “A Sketch Grammar ofWolaytta,” Japan Association for Nilo-Ethiopian Studies, 2014.

[47] L. A. Krukrubo, “The Data Science Methodology,” 2019. [Online]. Available: https://cognitiveclass.ai/courses/data-science-methodology-2/.

[48] I. Analytics, “Foundational Methodology for Data Science,” 2015.

[49] WORIDBIBLE.ORG, “The Bible of Wolaita,” *W*. [Online]. Available: http://worldbibles.org/language_detail/eng/wal/Wolaita.

[50] J. H. M. Jurafsky, Daniel, *Speech and Language Processing: An Introduction to Natural Language Processing, omputational Linguostics, and Spech Recognition*. PEARSON Prentice Hall,New Jersey, 2006.

[51] S. D. S. S. A. Basu, "A Hybird Model for part of speech tagging and its application to Bengali," vol. 1, 2004.

[52] E. L. Steven Bird, Ewan Klein, "Natural Language toolkit and Python," First Edit., 1005, Gravenstein Highway North, Sebastop: O'Reilly Media,Inc, 2009.

[53] T. Hariyanti, S. Aida, and H. Kameda, "Samawa Language Part of Speech Tagging with Probabilistic Approach : Comparison of Unigram , HMM and TnT Models," in *The 3rd International Conference on Computing and Applied Informatics,* 2019.

[54] I. M. Singh, Jyoti, Nisheeth Joshi, "Development of Marathi Part of Speech Tagger," *IEEE,* 2015.

[55] K. Rungta, "Learn Python in 1 Day," 2018. [Online]. Available: https://www.amazon.com/Learn-Python-Day-Complete-Examples-ebook/dp/B01IRGB6MY. [Accessed: 02-Apr-2020].

[56] B. Steven, "'Natural Language:the natural language toolkit,'" in *In: proceeding of CLOING/ACL on interactive presentation session, Association for computational Linguistics, Morristown,NJ,* 2006, vol. 1.

[57] E. Loper and S. Bird, "NLTK : The Natural Language Toolkit," *ResearchGate,* 2002.

[58] Techopedi, "Natural Language Toolkit(NLTK)," 2019. [Online]. Available: https://www.techopedia.com/definition/30343/natural-language-toolkit-nltk.

Appendix

Appendix A: Sample of Corpus

Ha/DD katamayi/NN gita/ADJ zal''iyan/NN ,/PUN gam''idi/VC minnida/VR wogan/NP ,/PUN keehi/ADJ aakki/ADV aakki/ADV biya/NV yeda/ADJ de'uwan/NP woyikko/CJ giigennabaa/NV haniyoogaaninne/VC hayimaanootiyaa/NN coratettan/ADV erettidaagaa/VV ./PUN Yesuusi/NN kiittidoogee/NV minttidi/VC odanau/VI koyidobati/VC ,/PUN he/DD woosa/NN keettaa/NN giddon/PRP de'iya/ADJ shaahotettaabaanne/NC yeda/ADJ de'uwaabaa/NP ,/PUN asho/ADJ gayitotettaabaanne/NC geluwaa/NN ehuwaabaa/NN ,/PUN qofa/NN giddo/PRP oyishatubaa/NN ,/PUN woosa/NN keettaa/NN wogaabaa/NN ,/PUN geeshsha/ADJ ayyaanaa/NN imotatubaanne/NC dendduwaabaa/VV ./PUN

Wonggeliyaa/NN mishiraachchoyi/NN ha/DD oyishatubaa/NP odiyoogaa/NV naagidi/VC shiishshees/VV ./PUN

Siiqoi/NP Xoossai/NN ba/PRP asaassi/NN immido/VR imotu/NN ubbaappe/ADVC aadhdhiyaagaa/NV gidiyoogaa/NV odiya/ADJ miraafe/NN tammanne/CNN heezzayi/AJN ha/DD maxaafan/NP keehippe/ADV daro/ADJ erettida/VR kifile/NN geetettidi/VC qofettee/VV ./PUN Taani/PP ,/PUN Xoossaa/NN sheniyan/NP Yesuus/NN Kiristtoosi/NN kiittidoogaa/NV gidanawu/VI xeesettida/VV PHauloosinne/NC ,/PUN nu/PP ishaa/NN sostteniisi ,/PUN Qoronttoosan/NP de'iya/ADJ Xoossaa/NN woosa/NN keettaassi/NP ,/PUN Yesuus/NN Kiristtoosan/NP geeyidaageetussi/VC ,/PUN etassinne/PC nuussi/PP Godaa/NN gidiya nu/PP Godaa/NN Yesuus/NN Kiristtoosa/NN sunttaa/NN ubba/ADV sohuwan/VC xeesiya/ADJ ubbatuura/NP geeshsha/ADV gidanawu/VI xeesettidaageetussi/VC ha/DD dabddaabbiyaa/NN xaafoos/VV. /PUN

Xoossai/NN ,/PUN nu/PP aawayinne/NC nu/PP Godayi/NN Yesuus/NN Kiristtoosi/NN inttessi/NN aaro/ADJ kehatettaanne/NC sarotettaa/NN immo/VV ./PUN

Xoossai/NN salotanne/NC sa'aa/NN koyiro/ON medhdhiis/VV ./PUN

He/DD wode/ADV sa'ai/NP giigiichchibeennanne/VC ayibinne/VC bainna/VR mela/NN ;/PUN ciimmaa/NN bollan/PRP xumai/NP de'ees/VV ;/PUN Xoossaa/NN Ayyaanayi/NP haattaappe/NP bollaara/PRP maayi/VC uttiis/VV ./PUN

Xoossai/NN hegaappe/DPRP guyyiyan/PRP hagaadan/VR yaagidi/VC haasayiis/VV ;/PUN poo'oi/NN hano/VR yaagiis/VV ;/PUN yaagin/VR poo'oi/NN haniis/VV ./PUN

Poo’oi/NN lo’’o/ADJ gididoogaa/NV Xoossai/NN be’iis/VV ;/PUN be’idi/VC poo’uwaa/NN xumaappe/NP shaakkiis/VV ./PUN Zakkaariyaasi/NN hegaa/DD be’idi/VC ,/PUN dagammidi/VC yayyin/VR kiitanchchayi/NN Zakkaariyaasa/NN hagaadan/DPRP yaagiis/VR ;/PUN yayyoppa/VR ;/PUN ne/PP woosayi/ADV siyettiis/VV ./PUN

Ne/PP machchiyaa/NN Elssaabeexa/NN neessi/PRPR attuma/ADJ na’a/NN yelana/VV ;/PUN neeni/PP he/DD na’aa/NN Yohaannisa/NN yaagada/ADV sunttana/VV ./PUN

I/PP yelettin/VC neeni/PP daro/ADJ ufayittana/VR ;/PUN haratikka/ADV ufayittana/VV ./PUN

Xoossaa/NN sinttan/NP i/PP gita/ADJ gidana/VV ;/PUN mattoyiyaabaanne/NC woyine/ADJ eessa/NN uyenna/VV ./PUN Biron/ADJ ba/PRP aayee/NN uluwan/NP de’ishin/VC ,/PUN geeshsha/ADJ Ayyaanayi/NN aani/PPRP kumana/VV ./PUN

Israa’eela/ADJ asaappe/NP daruwaa/ADJ godaakko/NP eta/PP Xoossaakko/NP zaarana/VR ;/PUN godaassi/NP asaa/NN giigissanawu/VI aawatu/NP wozanaa/NN naatukko/NP zaaranawu/VI azazettennaageetakka/NV xillotettaa/ADJ eraassi/NP zaaranawu/VI Eelaasa/NN ayyaanaaninne/NC wolqqan/NP godaappe/NP sinttaara/PRP bees/ADV yaagiis/VV ./PUN

Zakkaariyaasi/NN ,/PUN taani/PP cima/VR ;/PUN ta/PP machchiyaanne/NC ceeggaasu/VR ;/PUN hegaa/DD ta/PP ayibin/WHP eranee/VV ?/PUN

Kiitanchchayi/NN zaarettidi/VC yaagiis/VR ;/PUN taani/PP Xoossaa/NN sinttan/PRP eqqiya/VR Gabreela/NN ;/PUN taani/PP ne/PP sinttan/PRP eqqada/VC haasayanawu/VI qassi/CJ ha/DD mishiraachchuwaa/NN neessi/PPRP odanawu/VI tana/PP Xoossayi/NN neekko/PPRP kiittiis/VV ./PUN

Ha’’i/ADV siya/VR ;/PUN ba/PRP wodiyan/NP polettana/VC ta/PP qaalaa/NN neeni/PP ammanabeenna/VC gishshawu/CJ ,/PUN ta/PP neessi/PPRP odidoADJ/ yohoyi/NN polettanaashin/VC neeni/PP duudana/VV ;/PUN ne/PP doonayi/NN haasayanawu/VI danddayenna/VR yaagiis/VV ./PUN Kiitanchchatu/NN mazamuriyaabaa/NP ,/PUN Yesuusi/NN yelettido/VR wode/ADV henttanchchati/NN biidi/VC be’idoogaa/NV yootiya/ADJ taarikee/NN ,/PUN Yesuusi/NN naatettan/ADV de’iiddi/VR beeta/ADJ maqidasiyaa/NN biidobayi/VC qaretanchcha/ADJ saamiraawiyaanne/NC bayidi/VC beettida/VR na’aa/NN taarikee/NN luqaasa/ADJ wonggeliyaa/NN giddo/PRP xalaalan/ADV beettiyaageeta/VV ./PUN Ha/DD wonggeliyaa/NN giddon/PRP ha/DD gaxaappe/NP hini/DD gaxaa/NN gakkanaashin/VC waannatidi/VC xeelettidabati/NN woosaa/NN ,/PUN geeshsha/ADJ ayyaanaa/NN ,/PUN Yesuusa/NP oosuwaa/NN giddon/PRP maccaasaassi/NP de’iya/ADJ sohuwaanne/NC

Xoossayi/NN nagaraa/ADV attogiyoogaa/VV ./PUN Yihudaa/ADJ biittaa/NN kawuwaa/ADJ Heeroodisa/NN wodiyan/ADV Zakkaariyaasa/NN yaagiyo/ADJ issi/AJN qeesee/NN de'ees/VV ./PUN I/PP abiiya/ADJ qeesetuppe/NP issuwaa/VV ./PUN A/PP machchiyaanne/NC qeesiyaa/ADJ Aaroona/NN zare/NV ;/PUN i/PP sunttayi/ADV elssaabeexo/VR ./PUN

Eti/PP naa''ayikka/CNN Xoossaassi/NP xillo/ADJ gididi/VC godaa/NN higgiyaanne/NC azazuwaa/NN ubbaa/ADV moorennan/VR naagidi/VC polidosona/VV ./PUN

Elssaabeexa/NN mayine/ADJ gididogishshawu/CJ ,/PUN etassi/PPRP na'i/ADV baawa/VV ;/PUN naa''ayikka/CNN daro/ADV cimidosona/VV ./PUN Bantta/PRP qeesetu/NN kayan/NP Zakkaariyaasi/NN Xoossaa/NN sinttan/NP oottishin/VC ,/PUN qeesetu/ADJ meeziyaadan/NP ,/PUN godaa/ADJ beeta/ADJ maqidasiyaa/NN gelidi/VC ixaanaa/NN cuwayanawu/VI saamayi/ADV a/PP gakkiis/VV ./PUN Ixaanaa/NN i/PP cuwayiyo/VR wode/ADV asayi/NN ubbayi/ADV Karen/NN eqqidi/VC ,/PUN Xoossaa/NN woossees/VV ./PUNXoossaa/ADJ kiitanchchayi/NN ixaanaa/NN i/PP cuwayiyo/VR sohuwaappe/NP ushachcha/ADJ baggaara/PRP eqqidi/VC ayyo/PPRP beettiis/VV ./PUN Sa'ai/NP qammiisinne/VC wonttiis/VV ;/PUN hegeenne/DPRP oyiddantto/AJN gallassa/VV ./PUN Xoossai/NN hegaappe/DPRP guyyiyan/PRP hagaadan/VR yaagidi/VC haasayiis/VV ;/PUN Paxa/NN de'iya/VR dumma/ADJ dumma/ADJ meretati/NN haattan/NP darona/VR ;/PUN kafoikka/NP sa'aappe/NP bollaaranne/PRP salo/NN gufanttuwaappe/NP garssaara/PRP paallo/VR/PUN yaagiis/VV ./PUN

Yaatidi/VC Xoossai/NN abban/NP de'iya/VR gita/ADJ do'ata/NN ,/PUN haatta/NN giddon woxxi/VC qaaxxiya/VR meretata/NN ubbaanne qefiyaara/NP de'iya/VR kafuwaa/NN ubbaa a/PP qommuwan/NP qommuwan/NP medhdhiis/VV ./PUN

Medhdhidi/VC Xoossai/NN hegee lo''o/ADJ gididoogaa/NV be'iis/VV ./PUN

Xoossai/NN eta/PP ,/PUN /PUN Yelettite/VR ;/PUN corattite/VR ;/PUN abbaa/NN haattaa/NN kumite/VR ;/PUN kafoikka/NP biittaa/NN bollan/PRP coratto/VR/PUN yaagidi/VC anjjiis/VV./PUN Sa'ai/NP qammiisinne/VC wonttiis/VV ;/PUN hegeenne ichchashantto/ gallassa./PUN Xoossai/NN hegaappe/ guyyiyan/PRP hagaadan/VR yaagidi/VC haasayiis/VV ;/PUN Paxa/NN de'iya/VR meretati/NN bantta/VC qommuwan/NP qommuwan/NP ;/PUN meheti/NN ,/PUN biittaara/NP gooshettiya/VR meretatinne/NC do'ati/NN bantta/VC qommuwan/NP qommuwan/NP biittaa/NN bollan/PRP kiyona/VR yaagiis/VV ;/PUN yaagin/VC haniis/VV ./PUN Luqaasa/ADJ Wonggelee/NN Yesuusa/NN koyiro/ON yaanagi/VR wottidaagaa/NV ;/PUN Israa'eela/NN ashshiyaagaa/NV hegaadankka/DPRP asaa/ADJ naata/NN

ubbaa/ADV ashshiyaagaa/NV yaagidi/VR odees/VV ./PUN Luqaasi/NN Yesuusa/NN Xoossaa/NN Ayyaanayi/NN xeesidoogee/NV hiyyeesatuyyo/NP wonggeliyaa/ADJ mishiraachchuwaa/NN odanaassa/VR gidiyoogaanne/VC ha/DD wonggelee/NN asaassi/NP de'iya/VR dumma/ADJ dumma/ADJ metuwaa/NN qaariyo/VR wolqqaa/NN oyiqqidaagaa/NV gidiyoogaa/ADV odees/VV ./PUN Luqaasa/ADJ wonggeliyaa/NN giddon/PRP Yesuusabaa/NP odiya/VR koyiro/ON miraafetuuninne/NC qassi/CJ Yesuusi/NN saluwaa/NN pude/PRP kiyidoogaa/NV odiya/VR wurssetta/ADJ miraafetun/NP darissidi/VC odidoyi/ADV ufayissaabaa/VV ./PUN

Yesuusi/NN saluwaa/NN kiyidoogaappe/VC guyyiyan/PRP kiristtinnaa/ADJ ammanoyi/NN waani/WHP diccidaakkonne/VC waani/WHP aakkidaakkonne/VC odiya/VR taarikiyaa/NN ha/DD maxaafaa/NN xaafida/VR bitanee/NN Yesuusi/NN kiittidoogeetu/NV oosuwaa/NN maxaafan/NP odiis/VV ./PUN

Appendix B: Some of extracted Lexical Transformation rule

VR->VC if the text of the current word suffix is 'sonaanne'

ADJ->CNN if the text of the current word suffix is 'ntta'

NN→VV if the text of the following word is '.'

NN->VV if the text of the current word suffix is 'iis'

PRP->NP if the text of current word suffix is 'ssi'

NN→NP if the text of the preceding word is '.' And the text of the following word is ','

NN->AJN if the of the preceding word is 'tamanne'

NN->CNN if the text of the following word is 'ichchashu'

VV->PRP if the text of the preceding word is 'hara'

Appendix C: Some of extracted Contextual Transformation rule

NN->NP if the tag of the preceding word is 'PUN', and the tag of the following word is 'ADJ'

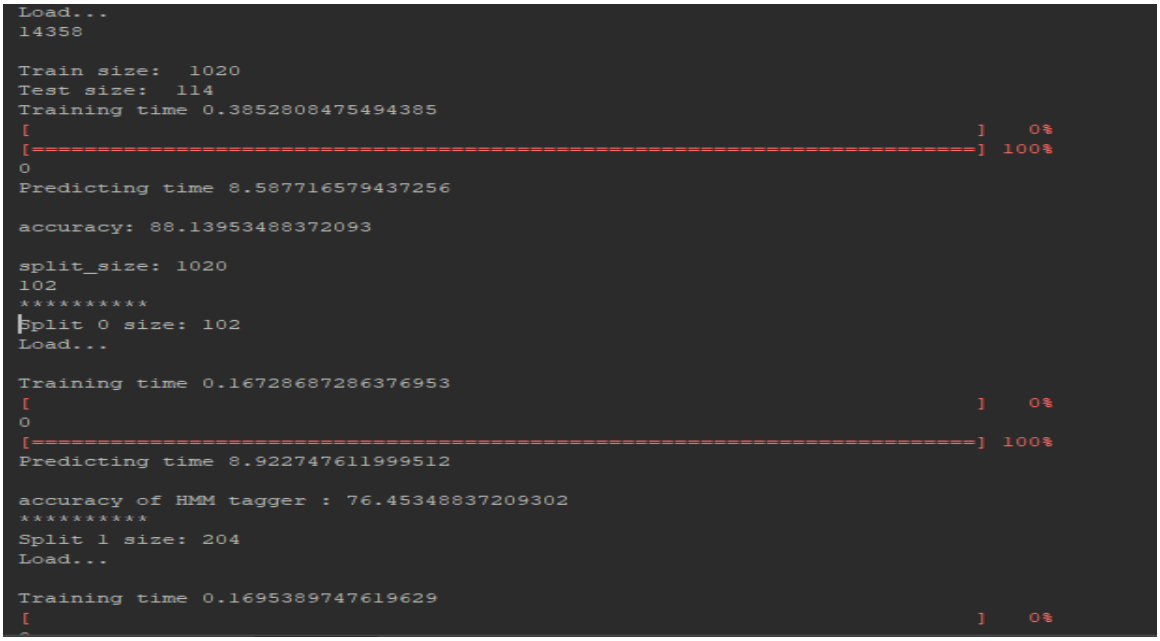
PC->PP if the tag of words i-3..i-1 is 'PUN'

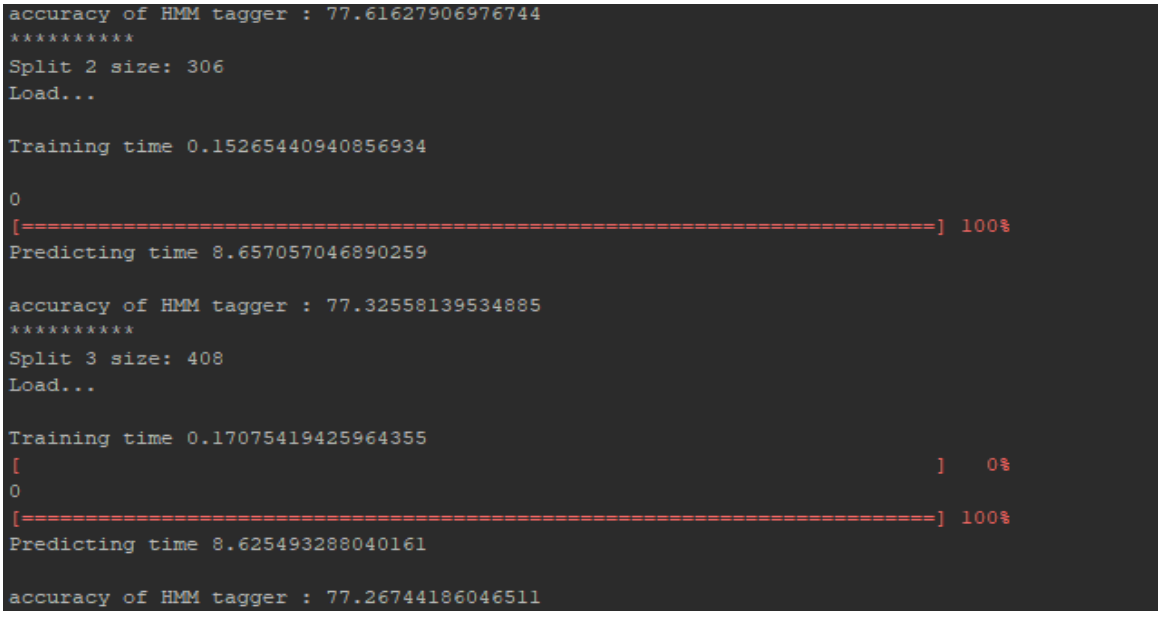
VV->NN if the tag of the preceding word is 'NN', and the tag of the following word is 'NN'

VV->VR if the tag of the following word is VV

Appendix D:

Sample Screenshots Performance result of HMM tagger, TBL tagger and Hybrid tagger





.

.

```
Split 9 size: 1020
Load...

Training time 0.36150097846984863

[============================================================================] 100%
0
Predicting time 8.505079984664917

accuracy of HMM tagger : 88.13953488372093

Process finished with exit code 0
```

Figure 10: Result of HMM Tagger

```
Run:  brill
C:\Users\bire\AppData\Local\Programs\Python\Python35\python.exe C:/Users/bire
Loading tagged data...
14358
[[('neeni', 'PP'), ('xoossaa', 'NN'), ('geeshshaa', 'NN'), ('gidiyoogaa', 'NV
TDS: [[('he', 'DD'), ('wode', 'ADV'), ('sa'ai', 'NP'), ('giigiichchibeennanne
total corpus: 1134
total words: 14358
Done loading.
Training Brill tagger on 1020 sentences...
split_size: 1020
102
**********
Split 0 size: 102
Training Brill tagger on 102 sentences...

Brill accuracy: 0.925581
**********
Split 1 size: 204
Training Brill tagger on 204 sentences...

Brill accuracy: 0.925581
**********
Split 2 size: 306
Training Brill tagger on 306 sentences...

Brill accuracy: 0.926744
```

Figure 11: Result of Brill Tagger Screenshot

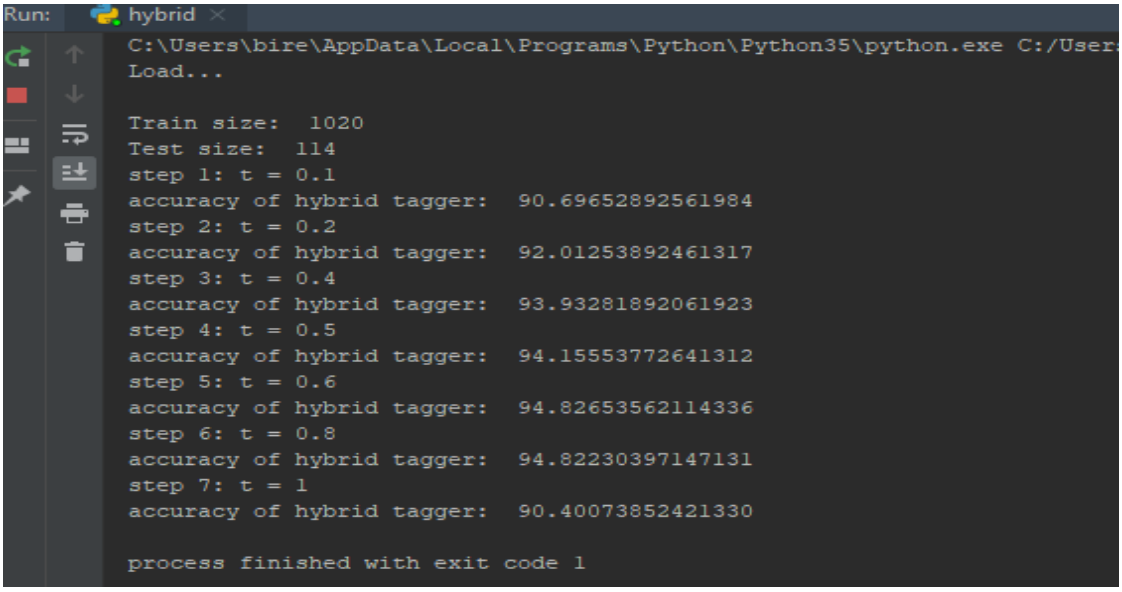


Figure 12: Result of Hybrid Tagger Screenshot