JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATICS

DEPARTMENT OF INFORMATION TECHNOLOGY

SEMANTIC ROLE LABELING FOR AFAAN OROMO SIMPLE SENTENCE

By: Dejene Bogale Balcha

A Thesis Submitted to the School of Graduate Studies of Jimma University in Partial Fulfillment of the Requirement for the Degree of Master of Science in Information Technology

Jimma, Ethiopia

December 2021

SEMANTIC ROLE LABELING FOR AFAAN OROMO SIMPLE SENTENCE

Dejene Bogale Balcha

Advisor: Getachew Mamo (Ph.D.)

This is to certify that the thesis prepared by Dejene Bogale, entitled as "Semantic role labeling for Afaan Oromo simple sentence" and submitted in partial fulfillment of the requirements for the Degree of Master of Science in Information Technology complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Examining Committee: -

Getachew Mamo (Ph.D.)		24/11/2021
Major Advisor	Signature	Date
Gashaw Zemene (MSc)		23/11/2021
Co-Advisor	Signature	Date
Worku Jifara (Ph.D.)		07/12/2021
External Examiner	Signature	Date
Zarihun Olana (MSc)		07/12/2021
Internal Examiner	Signature	Date
		07/12/2021
Chairperson	Signature	Date

DECLARATION

This thesis is a summary of my original study findings. Wherever other people's contributions are involved, every attempt is taken to make this known, with suitable citations. I, the undersigned, declare that this study has not been presented for a degree in any other university.

Dejene Bogale Balcha

Name of Student

Signature

15/12/2021

Date

ACKNOWLEDGMENTS

All of those who have assisted me in completing this work deserve my deepest gratitude.

First, above all I want to express my gratitude to God for providing me with the strength and determination to finish this study.

Second, I'd like to thank my adviser, Getachew Mamo (Ph.D.), for his generous and beneficial advice, fatherly support, direction, constructive ideas, and constant desire to assist me, as well as for his critical review of my thesis and constructive comments. Mr. Gashaw Zemene (MSc), my co-advisor, deserves special mention for all of his willingness and input in helping me finish my study.

Finally, I'd like to express my gratitude to my family and friends for their patience and support throughout this journey.

ABSTRACT

Natural language is used to recode human knowledge. Data is stored in computers or on paper in order to be processed and recorded for future use. Semantic role labeling (SRL) is one of the essential problems in the field of natural language processing (NLP) and its task was to determine the semantic roles (such as AGENT and PATIENT) of each argument that corresponds to each predicate in a phrase automatically. SRL is useful shallow semantic representations, and it is a very important intermediate step for several NLP applications, such as Information Extraction, Question Answering and Machine Translation. Traditional methods for SRL are grounded on parsing output, and require much feature engineering. So, the goal of this study was to develop a semantic role labeller for Afaan Oromo text using a deep learning algorithm. To solve Afaan Oromo SRL problems, we employ deep neural networks with long-short term memory (LSTM) and bidirectional long-short term memory (BLSTM). For this study 1800, Afaan Oromo simple sentences were used to train the system, which was semantically annotated with semantic role label using the BIO Tagging procedure and the PropBank annotation framework's principles. The experiment conducted by using 90%, and 10% of the total dataset for training and testing respectively. Experimental results show that the BiLSTM performed better and achieving the better results in terms of accuracy (80%), precision (81%), recall (80%), and f-measure (80%) as compared to LSTM in terms of accuracy (76%), precision (78%), recall (76%), and f-measure (76%). Based on experimental analysis, concluding remarks and recommendations are forwarded.

Keywords: Afaan Oromo, Semantic role labeling, Deep learning, Deep neural networks.

Table of Contents

List of figures

List of tables

Acronyms and Abbreviations

ARGM	Argument
AOSRL	Afaan Oromo semantic role labeling
ANN	Artificial Neural Network
BLSTM	Bidirectional Long Short-Term Memory
BGD	Batch gradient descent
CRF	Conditional Random Field
CBOW	Continuous Bag-of-Words
ConLL	Conference on natural language learning
ELMo	Embeddings from Language Models
FN	False Negative
FP	False Positive
LISA	Linguistically-informed self-attention
LSTM	Long Short-Term Memory
LOOCV	Leave-one-out cross-validation
ML	Machine Learning
MLP	Multi-Layer Perceptron
MBL	Memory Based Learning
NLP	Natural Language Processing
POS	Part of Speech Tagging
ReLU	Rectified Linear Units
RNN	Recurrent Neural Network
SRL	Semantic Role Labeling
SVM	Support Vector Machine
SGD	Stochastically Gradient Descent
TIMBL	Tilburg memory-based learning
TP	True Positive

CHAPTER ONE

1 INTRODUCTION

This chapter discusses about basic definition of semantic role labeling, problem statement, objective and methodology.

1.0 Background

Semantics is a branch of Natural Language Processing (NLP) that deals with determining the meaning of a statement. Semantic role labeling, also known as shallow semantic parsing, conducts the first steps in extracting meaning from a text by assigning generic labels or roles to words of the text, assuming that the meaning of this particular set of labels is understood by the computer [1].The task of recognizing arguments for a predicate and providing semantically meaningful labels to them is known as semantic role labeling [2].The main verb in a phrase is referred to as the predicate in linguistics. So, the goal of Semantic Role Labeling (SRL) is to figure out how these arguments relate to the predicate semantically. Because one of the most important aspects of learning natural language is comprehending events and their participants, semantic roles are symbolic entities that define the function of the participants in an event from the perspective of the situation in the real world. They make it possible to determine who did what to whom, when, when, and how.

Semantic roles are representations of the abstract role that a predicate's arguments can play in the event. SRL is the act of assigning labels to words or phrases in a sentence that represent their semantic role in the sentence, such as agent, patient, goal, or result, it also known as shallow semantic parsing or slot-filling in natural language processing [1]. SRL involves identifying the semantic arguments linked with a sentence's predicate or verb and categorizing them into their respective roles.

For example, given a sentence like "**Tola sold the book to Chala**", the task would be to recognize the verb "**to sell**" as representing the predicate, "**Tola**" as representing the seller (**agent**), "**the book**" as representing the goods (**theme**), and "**Chala**" as representing the **recipient**. By these ways we express the likely relationship to the syntactic role of the argument in the sentence and can represent general semantic properties of the arguments. ***Agents*** tend to be the subject of an **active sentence**, ***themes*** the **direct object**, and so on; in databases these relations are organized like proposition Bank (PropBank) and FrameNet electronic resource.

In recent times shown that in natural language processing like ,in question answering semantic role representations have many potential applications [3] [4], textual entailment [5], machine translation [6] [7], dialogue systems [8] and event extraction [9] and among others. From the point of view of the situation in the real-world semantic roles are symbolic entities that describe the function of the participants in an event [10]. It acts as a shallow meaning representation that can let us make basic deductions that aren't conceivable from the pure surface string of words, or indeed from the parse tree, that's the most reason that computational frameworks utilize it. In machine translation, this shallow semantics could be useful as an intermediate language. This sort of interpretation blunders regularly comes about in basic mistaken assumptions of the essential meaning of the initial input dialect sentences who did what to whom, for whom or what, how, where, when, and why, shallow semantic parsing can move forward interpretation exactness of static machine interpretation models is illustrated by [6] for the first time.

The contribution of SRL to open-domain factoid question answering is evaluated by [3] and the authors present a graph-based answer extraction model which effectively incorporates FrameNet style semantic role information and he achieves promising results. The performance gains over a shallow semantic parser trained on the FrameNet annotated corpus was demonstrated by authors. Those and other study work showed that SRL gives promising result for different NLP applications. Recently, supervised machine learning approach has been used to describe what counts as a predicate, define the collection of roles used in the task, and explicitly offer training and test sets for SRL.

1.1 SWOT Analysis

SWOT analysis (strengths, weaknesses, opportunities, and threats) allows researchers to assess the present state of the art, which is helpful in providing actionable insights.

Strength

It is beneficial for improving weaknesses and threats of existing system and shows areas to be confidently considered in proposed study.

1. Availability of huge amount of Afaan Oromo electronic data.
2. To use and manipulate the product of this study, no need of special keyboard, since Afaan Oromo uses Latin based script.

Weakness

List of weaknesses identified for productive completion of the study are:

1. The dataset has a big impact on the deep learning algorithms used for NLP. However, for the development of semantic role labeling for Afaan Oromo and other Afaan Oromo text processing NLP studies as a reference for SRL, there is a lack of a consistent and publicly available dataset.
2. Lack of clearly identified approach to annotate Afaan Oromo text with semantic role.
3. There is no previously develop model of Afaan Oromo NLP studies.

Opportunities

It demonstrates the chances from which the new study can be benefited.

1. Availability of deep algorithms that can be used to processes Afaan Oromo text.
2. Use of Afaan Oromo data found on online social media, generated every day by users with variable writing skills, different format and size.

Threat

Indicate challenges created by unconcerned development of semantic role labeler for Afaan Oromo that may lead to worsening the performance of final result of the study.

1. The accuracy of the model may vary on tuning of different model parameters.
2. Training data may comprehend incorrect labeled of the argument, which is main reason for performance degradation of the result.
3. It is difficult to assessing the understanding of Afaan Oromo text meaning skills of individuals, if everyone relies only on semantic role labeling.

1.2 Motivation

SRL has received considerable interest in the past few years [11] because of its significant contribution for different natural language understanding applications such as information extraction, question answering, machine translation, summarization, co-reference resolution [12], etc. Currently such types of natural language understanding applications were developed for different languages including Afaan Oromo using different technique [13] [14] [15].When SRL is used as an intermediate step for such application, it significantly improves their performances. At present-day, Afaan Oromo is official language of Oromia regional state. It is serving as a medium of instructions in primary schools, teacher training institutions, colleges and these days it is found to be a field of study in higher educational institutions. As a result, availability of Afaan Oromo textual information such as advertisements, educational resources, newspapers, posts and document on social media networks is highly increasing from day-to-day. However, when we need to processes those data meaning we need away to extract meaning and use for our intended task is not available, except parsing but SRL is one solution so, this situation has motivated us to develop SRL system for Afaan Oromo text.

1.3 Statement of the Problem

Afaan Oromo is a prominent African language that is widely spoken and utilized in most of Ethiopia, as well as some sections of neighboring countries such as Kenya and Somalia [7]. It is the official language of the state of Oromia at the moment (which is the largest Regional State among the current Federal states in Ethiopia). It is used by Oromo people, who are the largest ethnic group in Ethiopia. With regard to the writing system, “Qubee” (a Latin-based alphabet) has been adopted and become the official script of Afaan Oromo since 1991. Many domain areas produce large content of textual information in Afaan Oromo both in printed and digital format. Today as technology improves, several forms of information are in use everywhere in the World. Books, journals, articles and other documents can be accessed electronically.

Even though Lack of natural language processing tools and techniques that understand Afaan Oromo text such as part of speech tagging, morphological analyzers, question and answering, parsing and machine translation were developed but it difficult to understand the meaning of Afaan Oromo text by those application fully, this the major problem for many people that use the language as their means of information processing and understanding the event.

We have to parse Afaan Oromo sentences syntactically and semantically using constituent or dependency parsing in order to comprehend and analyze them. However, determining semantic roles solely based on syntactic interactions is problematic. This type of comprehension is more than just a matter of syntax. Parsing, on the other hand, isn't utterly worthless because it's a characteristic of SRL. A parse tree is used in a typical SRL pipeline to identify the predicate arguments. However, if the parse tree is incorrect, SRL performance may suffer. This has prompted SRL attempts to solve the parsing difficulty. This has motivated SRL approaches overcome problem of parsing. Due to its importance in understanding the semantics of natural language processing, various semantic role labeling systems for money languages have been created in the past [7]. However, none of the existing methods/systems are dealing with Afaan Oromo language, so there is no semantic role labeling system for Afaan Oromo. With the fact that the language is used in offices, schools and media, there is a huge electronic data available that encourages studies related to NLP tasks associated with the language. So, the development of SRL applications for this language is required to cope up with the current technologies of NLP.

Although SRL systems have proved to function relatively well in various controlled tests to date, future SRL studies faces a number of significant hurdles. It's still unknown what level of syntax is required to support strong semantic role analysis, and to what extent enhanced SRL performance is limited by current state of the art tagging and parsing technology. Beyond syntax, the relationship between semantic roles and additional semantic knowledge (such as WordNet, named entities, or even a catalog of frames) has received little attention in the current SRL model design [10]. A better understanding of these issues could aid in the development of approaches that are more generalizable and less reliant on enormous amounts of role-annotated training data.

Indeed, the requirement of most SRL techniques for such training data, which is both difficult and expensive to acquire, is the primary impediment to SRL's general adoption across genres and languages. In view of the performance degradation when a supervised system is confronted with unseen events or a test corpus deviating from training, this is a major obstacle for increase the use of SRL also in English, a language for which two large annotated corpora are available. It is critical for SRL's future success that research expands to include a deeper look into how to improve SRL's performance even when testing data differs from training data that must be addressed [16].

Aside from these open research questions, there are also methodological concerns about how research is conducted and assessed. Shared task frameworks have been critical in the development of SRLs because they allow for explicit comparisons of techniques, however such benchmark testing can focus research efforts too narrowly on small improvements in specific assessment measure. Improving the entire SRL approach in a significant way may require more open-ended investigation and more qualitative analysis (i.e., need further study to overcome the problem of standard framework (methodology) that help to evaluate the SRL work in every domain, language).

When SRL is used as an intermediate step for application develop in the past for different language including Afaan Oromo, it significantly improves their performances. Although a number of algorithms have been proposed for automatically assigning semantic roles to other languages, to the best of my knowledge, there is no prior work on automatic SRL for Afaan Oromo. Hence, this study fulfills the gap and address the issue of automatic sematic role labeler for Afaan Oromo simple sentence.

1.4 Research Questions

At the end of this study the following question is answered.

1. What are the challenges in the development of semantic role labeling for Afaan Oromo using deep learning?
2. How to label sentence constituent with semantic role label in Afaan Oromo text?
3. To what extent does deep semantic role labeler can help to labeling of Afaan Oromo sentence constituent with their semantic role?

1.5 Objectives

1.5.1 General Objective

The general objective of this studies is to design automatic semantic role labeling for Afaan Oromo simple sentences.

1.5.2 Specific Objectives

The specific objectives of the studies are: -

- ✓ To review literature on the techniques of automatic semantic role labeling so as to have a conceptual understanding of the area and identify the state-of-the-art in the area.
- ✓ To review related literature on sentence types and structures of Afaan Oromo language.
- ✓ To prepare semantically annotated sentence to train and test the model.
- ✓ To design Afaan Oromo semantic role labeling (AOSRL) for Afaan Oromo.
- ✓ To implement Afaan Oromo semantic role labeling (AOSRL) for Afaan Oromo.
- ✓ To test the performance of the model.
- ✓ To draw conclusions and suggest recommendations based on the output of the study for future work.

1.6 Scope and Limitations of the Study

The scope of this research work is limited to develop SRL for simple Afaan Oromo sentence and the study is limited to labeling of sixteen semantic role class. We deal only with data in textual format without any kind of grammar and spelling correction (i.e., the grammar checking and spell correction only done during annotation of Afaan Oromo sentence with semantic role). Since Luck of sufficient documents on the background development of automatic semantic role labeling system for Afaan Oromo texts and due to the scarcity of local reference materials near the background of the study, we did not cover all Afaan oromo sentence.

1.7 Application of Results

This research work can be one of the intermediate components of higher-level NLP applications. As a result, researchers working to improve computers' ability to process Afaan Oromo language may profit from the findings of this study, as the system plays a critical role in many areas of NLP for Afaan Oromo language. Researchers working in the fields of machine translation, information extraction, question answering, and text summarization, in particular, will benefit from this study. Because semantic content interpretation of the text is one of the primary performance issues in such systems. Since one of the major performance challenges in such type of applications are semantic content understanding of given text. Therefore, Semantic Role Labeling significantly improve their performance by simplify this problem in Afaan Oromo after the completion of this study.

1.8 Thesis organization

The rest of this document is organized as follows. Literature review in AOSRL is explained in Chapter Two. Design of SRL for Afaan Oromo sentence was discussed in Chapter three. The Experimentation Results, Discussions and Evaluations are discussed in Chapter four. Finally, Chapter five presents conclusions from experimental observations and future works to show further areas of improvement on AOSRL systems.

CHAPTER TWO

2 LITERATURE REVIEW

2.1 Introduction

In this chapter, we discuss about Afaan Oromo sentence formation to have conceptual overview of Afaan Oromo sentence structure and its type and related literature review in the semantic role labeling is presented to enlightens briefly on some of the work done by those researchers on both local and foreign language's semantic role labeling researches and state-of-the-art approaches are briefly discussed.

2.2 Afaan Oromo Sentences

A sentence is a comprehensive collection of words or phrases that conveys a statement, inquiry, exclamation, or command and usually includes a subject and predicate. A phrase is a fundamental component of a sentence. A phrase is a combination of words that stand together. It is a syntactic structure that is wider than a word and smaller than a sentence. Phrase can be constructed only from a head word or a head word combined with other words or phrases. When a phrase is constructed from a head word and other words or phrases, the other words or phrases can be specifiers, modifiers or complements.

Specifiers: -are words used to specify the identity, location, number, possession, etc of the head word. Specifiers can be either primitive or derived. For example, in the sentence "kun kitaaba Tolaati**"** (This book is belonging to Tola) the specifier that shows the owner of the book.

Modifiers: Modifiers play an important part in determining the meaning of a word in Afaan Oromo. In Afaan Oromo modifiers can occur before the target word (the word it modifies or describes). It adds detail or limits or changes the meaning of another word or phrase [17]. It was used to indicate the amount, time, place, type, etc. of the head word or phrase.

Adjectival phrase, noun phrase, prepositional phrase or sentences can be considered as modifiers. For example, in the phrase" konkolaataa adii" (white car), the word "adii" (white) indicates what type of color the car has.

Complements: -are words or phrases that are used to make the ideas complete. For example, in the sentences “Daabbo nyaadhe” (I ate bread) and “Daabbo qamadii nyaadhe” (I ate wheat bread), the first sentence does not give complete information about the bread but in the second sentence “Daabboo qamadii” (wheat bread) is a complement that indicates from what the bread is made.

The sentence structure in Afaan Oromo is subject-object-verb (SOV). SOV is a sentence structure where the subject comes first, then the object and the verb next to the object. For example, if we take Afaan Oromo sentence Dajaneen nyaata nyaate**, “**Dajaneen” is the subject, “nyaata” is the object and “nyaate” is the verb of the sentence. In case of English, the sentence structure is subject-verb-object. For example, if the above Afaan Oromo sentence is translated into English equivalent to “Dejene ate food” where “Dejene” is the subject “ate” is the verb and “food” is the object, however, The subject-object-verb (SOV) format is used by Afaan Oromo. However, nouns alter depending on their function in the sentence, and word order is flexible, however verbs usually follow their subjects and objects. In most cases, indirect objects come after direct objects.

For any language the types of the phrases are determined by the categories of words in a language. Therefore, since there are five-word categories in Afaan Oromo language, there are five phrase classes which are noun phrase (NP), verb phrase (VP), adjectival phrase (JJP), adverbial phrase (ADP), and Ad positional Phrase (APCP). A sentence is a complete expression that has a subject and predicate and conveys a statement, inquiry, exclamation, or command. For example, when we see the following phrases: ‘Mana citaa ‘(thatched house), there are two nouns, which make the noun phrase: Mana (house) and citaa (thatched)- Noun phrase [18]. Inni kara manaa deeme (He went to the house)-Ad positional phrase, **‘**Mana Dejene’ (Dejene’s house) or verbs as adjectives like ‘Farda adii” (white horse)- Adjectival Phrase**, ‘**kaleessa galgala’ **(**yesterday night)- Adverbial Phrase and **‘**Inni dhufe’ (He came)-Verbal Phrase.They do not convey full meaning when they are spoken in separate, instead they raise questions like what did? who did? and what has done? However, when these phrases are combined together, they form a complete sentence and answers all the above questions.

From the structural point of view, sentences are basically constructed from noun phrase and verb phrase. The structure of a sentence can be either simple or complex based on the number of verbs it contains.

2.2.1 Simple Sentence

Simple sentences in Afaan Oromo are sentences, which contain only one verb and that can have a full meaning. Simple sentence can be constructed from NP followed by VP, which only contain single a verb. A simple sentence consists only one main verb. For instance:

- ✓ Tolan burcuqqo cabse (Tola broke the glass)
- ✓ Caalaan dhufe – "Chala came"
- ✓ Abdin haableedhaan hoolaa qale (Abdi slaughters the sheep with the knife)
- ✓ Dajaneen obbolessaf qarshii erge (Dejene sends money to his brother)

Because each of the above sentences is constructed by using a single verb" cabse "(breaks)," qale "(slaughters) ፣ "erge"(sends)", the sentence is called simple sentence. In previous study, in order to know whether the noun phrase is subject or object and the propositional phrase is modifier in a sentence, first we must show the position in the parse tree then we must show the morphological reflection. We can call the two features structural and morphological, the best one is morphological criterion because all noun phrases that came at start of a sentence may not be subject. Simple sentence is classified into four kinds called declarative sentence, interrogative sentence, imperative sentence and negative sentence.

Declarative sentences (Hima Addeessaa yookin Himaamsa)

A simple sentence can explain about a subject and the explanation may be about the state of beingness, change or assimilation such sentences are called declarative sentences [37]. For example: In contrast to command, question or exclamation, if the sentence is a statement, it is declarative sentence. It is always ended with the period(.) mark, which is the same in English and equivalent to (.) in Afaan Oromo. They are used to convey information. Declarative sentences can be positive or negative sentences. Negative sentences simply negate a declarative statement made about something.

Example3:

- ✓ Abdiin barataadha(abdi is student) .
- ✓ Caalaan dhufe (Chala came).
- ✓ Abdiin loltuu tahe(Abdi becomes a soldier) .
- ✓ Senaan hadha ishee fakkati (Senaa looks like her mother).
- ✓ Tolosaan mana barumsaa hin deemne(Tolosa did not go to school).

Examples 1 and 3 explain about what Abdi is (the state of beingness). When we say *Abdi* becomes a soldier in example 3, we want to show the change from not being a soldier to becoming a soldier. Example 4 shows similarity between *Senaa* and *her mother*, in example5, the sentence is negative declarative sentence. The verb hin deemne ‘did not go" is negated by the prefix hin- ‘not’.In Afaan Oromo language grammatical voice of the verb can be active or passive. They are defined by how they are used in relation to the subject

Active Voice

In active voice, subject of the sentence is the doer of an action.

- ✓ Barataan dabtara isaa cicire (The student tears his exercise book)
- ✓ Barsisaan barattotaaf kitaaba kennef (The teacher gives a book for his students)

The examples clearly indicate as the subjects of the sentences perform the action expressed in the verb on others. For example, in the sentence" Barataan dabtara isaa cicire" (The student tears his exercise book) the action of tearing is done by Barataan (the student) the receiver of the action “dabtara isaa" (his exercise book). As a result, the subject “Barataan" (the student) is doer of the action and the object “dabtara isaa" (his exercise book) is receiver of the action.

In the second example, the action of giving is done by the subject “barsisaan"(the teacher) on the object “kitaaba" (book). But there are also receivers of the book. In this kind of sentence 3 main parts (subject, direct object and indirect object) are mandatory and the noun phrase which receives an action “kitaaba “(book) becomes direct object and the prepositional phrase “barattotaaf"(for his students) become indirect object.

- ✓ Chalaan gara gabayaa deeme(Chala goes to market).
- ✓ Ijollen mukarra korte(The children climb on the tree).

The first example indicates as the subject of the sentence goes from one place to the other and the action of going is shown by the verb “deeme” (goes). These kinds of verbs need complement (prepositional phrase). So, the prepositional phrases “gara gabayaa” and “mukarra” (to market and on the tree) are used in the given examples. For example, the verb “deeme” (goes) is joined with prepositional phrase “gara gabayaa” (to market) and form the sub verbal phrase. Then it is connected with the subject “Chaalaan”(chala) to form the sentence Chaalaan gara gabayaa deeme (chala goes to market) and answers the question “Where does Chala go?”.

- ✓ Ijollen balbalarra dhaabbatan (The children stand at the door).
- ✓ Totltun sirerratti kufte (Toltu falls on the bed).

A sentence like “Ijollen balbalarra dhaabbatan” (The children stand at the door) can show where the subject performs a specific activity and the place is indicated by the prepositional phrase used in the sentence. For example, in the above sentence the children perform the action of standing at the door, so “balbalarra” (at the door) shows where the subject of the sentence performs the activity. Therefore, the sentence that talks about the doer of an action is called active sentence.

Passive Voice

In passive voice a noun phrase that receives an action can be used as a subject of a sentence.

For example:

- ✓ Chaaltun mudamteerti.
- ✓ Bunni tumameera.

In the examples, the subjects “Chaaltun and Bunni” are the receivers of the action not the doers. The doer is not clearly indicated. However, in the sentence” Tolaan Chaaltu mudeera” the subject “Tolaan” is a doer and the object “Chaaltu” is a receiver. Then, we can understand that the subject of a sentence in passive voice can be the object of a sentence in active.

Interrogative sentences (Hima Gaaffii)

In Afaan Oromo Interrogative sentences are sentences that can form a question. The question can be the one that asks the known thing to be sure or the one that asks the unknown one. These types of sentences always end the question mark which is symbolled as?’and sentences consist of interrogative pronouns, eenyu? ‘who’, yoom? ‘when’, maal? ‘what’, meeqa?, eessa? ‘‘were’’, etc. Example: Guyyaan har’aa maali? ‘What is the day of today?’.

Imperative Sentences (Hima Ajajaa)

Imperative sentences can be used when someone wants to pass instruction or commands, most of the time, the subjects of imperative sentences are second person pronouns. However, when the command is passed for the third person, the subject of the sentence can be third person pronouns or nouns. Example: Hojii manaa hojjadhu. ‘Do homework.’
The subject is ‘you’ second person for both feminine and masculine singular and plural.

Exclamatory Sentences (Hima Raajeffannoo)

A type of simple sentence that expresses strong feelings (excitement or emotions) by making an exclamation in Afaan Oromo is called, an exclamatory sentence is (Compare with sentences that make a statement, express a command, or ask a question). Exclamatory sentences are rarely appearing in academic writing, unless they're part of quoted material. Example: ajaa’iba kuni**!** **‘**This is a surprise’.

2.2.2 Complex Sentences

Complex sentence in Afaan Oromo grammar is formed from either complex noun phrase or complex verb phrase or both. In other words, a complex sentence can have a complex NP and a simple VP, a simple NP and a complex VP or both complex NP and complex VP. Complex NPs contain at least one embedded sentence, which can be complemented or other type phrase. On the other hand, complex VPs contain at least one sentence or more than one verb.

Based on this we can have complex sentences which is constructed from one independent clause and one or more dependent clause. And also, which has one or more independent clause and two or more dependent clauses, this called compound complex sentences.

Example: - ‘Yoo dhufuuf taate, ganamaan koottu.’

‘Qilleensarras kaattu, lafarras arreeddu walgeettin teenya Finfinnee dha.’

2.3 Semantic Role

NLP is a new interdisciplinary field variously called computer speech and language processing or human language technology or computational linguistics. It is the computerized approach to analyze text based on both set of theories and set of technologies [16]. The main goal of this field is to make computers perform useful tasks concerning human language, such as enabling human-machine communication, improving human-human communication, or simply doing useful processing of text or speech [16].Studying the complex language behavior needs good understanding of the language (level of linguistic analysis). For instance,

- ✓ **Phonetics and Phonology**: Knowledge about linguistic sounds.
- ✓ **Morphology**: Knowledge of the meaningful components of words.
- ✓ **Lexical**: Knowledge about lexical meaning of words and parts of speech analysis.
- ✓ **Syntax**: Knowledge of the structural relationships between words (grammar and structure of sentence).
- ✓ **Semantics**: Knowledge of meaning of a word, phrase and sentence.
- ✓ **Pragmatics**: Concerned with the purposeful use of language in a situation and utilizes context over and above the contents of the text for understanding.
- ✓ **Discourse**: Knowledge about linguistic units larger than a single utterance, i.e., deals with the properties of the text as a whole that convey meaning by making connection between component sentences.
- ✓ **Disambiguation**: Refers to the resolution of ambiguities that occur at different levels of language analysis.

It may involve all or some of these levels of analysis in natural language processing task. Our study focuses on the area of semantic. We deal with one type of semantic annotation that is called semantic role labeling. When neither the raw linguistic inputs nor any of the structures created from them by any of the transducers support the type of semantic processing that is needed, meaning representations are required. To execute tasks involving the meaning of linguistic inputs, we need representation that bridges the gap between language input and non-linguistic knowledge of the environment [3]. As a result, semantic role labeling is an effective means of achieving this goal.

A semantic role depicts the link between a syntactic constituent and a predicate (argument and predicate). It specifies the role of a verbal argument in the event denoted by the **verb**: **agent, a patient, an instrument**, and so on. It is an abstract model of the role an **argument** plays in the **event** described by the **predicate** [3]. This can be discussed in three different levels of generality:

- ✓ Verb-specific semantic roles

  E.g., runner, killer, speaker, broker, etc.
- ✓ Thematic relations: which are generalizations across the verb-specific roles.

  E.g., agent, instrument, experiencer, theme, patient
- ✓ Generalized Semantic Roles: are generalizations across thematic relations.

The task of automatically determining the semantic role of each argument of each predicate in a phrase is known as semantic role labeling [19]. It allows us to define some of an argument's semantics in connection to the predicate and corresponds to the well-known notion of lead: "WHO did WHAT to WHOM, HOW, WHEN, and WHERE" [20]. It begins the process of extracting meaning from text by assigning generic labels or roles to the text's terms. The role labeling task typically entails identifying the parts of each target predicate (argument identification) and assigning semantic responsibilities to them (argument classification). However, in order to recognize and classify these arguments, SRL systems must first identify the target predicate (predicate identification) and then assign it a sense number (predicate classification). Apart from the role labeling information, the input data comprises multiple levels of annotation such as Pos tags, chunks, named entities, and parse trees.

2.3.1 Common List of Semantic Roles

Arguments are classified into two categories based on their semantic roles: (a) required arguments, which reflect central players in an event, such as agent, patient, topic, and instrument; optional arguments (adjuncts), which are optional for an event but provide additional information about it, such as location, time, and method [21].

Agent: Initiator of action, capable of volition, deliberately performs the action.

Example: [Ethiotelekom]***agent*** ummata harka qalleyif gargaarsa godhe .

Patient: Affected by action, undergoes the action and has its state changed.

Example: Hojjettonni [dallicha]***patient*** jijjigsan .

Experience: Entity moving or being "located" receives sensory or emotional input.

Example: Ijaarsi hidha haromsaa [ummata Ethiopiyaa] ***Experiencer*** gammachise.

Theme: Entity moving, or being "located", undergoes the action but does not change its state. Sometimes used interchangeably with patient.

Example: -Dhaabbatichi [qorannowwan 67] ***theme*** xumure.

Instrument: Used to carry out the action,

Example: Tolaan[furtudhaan]***instrument*** mana banate.

Location: Where the action occurs.

Example: Omishini gaazii boba'aa [nanno somaletti] ***location*** argame .

Direction or Goal: Where the action is directed towards.

Example: Dhaabbatichi omisha isaa [gara baha giddugalessa] ***direction*** erge.

Source: Where the action originated.

Example: Qonnan bulaan omisha isaa [maasirrra]***source*** gara gabayaa gesse .

Time: The time at which the action occurs

Example: Pirojectiin Shaggara[har'a] ***time*** ebbifame.

Beneficiary: The entity for whose benefit the action occurs.

Example: Yunivarsitin Finfinne [Artis Alii Birratif] ***Beneficiary*** Doctereta kabaa kennef.

Manner: The way in which an action is carried out.

Example: Dhibeen koranaa [saffisaan] ***manner*** dabalaa jiraachuu ibsame.

Purpose: The reason for which an action is performed

Example: -Ministeerri Fayyaa [midhamtoota dhibee buusaa hirdhisuf] ***purpose*** sosochii jalqabe.

2.3.2 Challenge in Semantic Role Labeling

Semantic role assignment is crucial to develop an efficient NLP system. But, despite this potential benefit, it has proved very difficult to come up with a standard set of roles, and equally difficult to produce a formal definition of roles like AGENT, THEME, or INSTRUMENT [16]. Consider the AGENT role: most AGENTS are living, volitional, sentient, and causal, but a single noun phrase may not have all of these characteristics.

2.3.3 Component for Sematic Role Labeling

Automatic processing of semantic roles is a relatively new subject of study [16]. It relies strongly on the experience and results achieved in other domains of automatic text analysis such as morphological and syntactic, which provide linguistic knowledge about the text that is necessary for this task.

2.3.3.1 Part of Speech Tagger

POS, also called grammatical tagging or word category disambiguation, method of labeling of each word of a written text with an appropriate part of speech tag. It means POS assigns whether a given word is used as a noun, adjective, adverb, verb, etc. A group of words with comparable grammatical features is referred to as a part of speech. Words that are assigned to the same word part of speech generally display similar behaviour in terms of syntax.

POS tagging can be used as an intermediate step for higher level NLP tasks such as parsing, Text to Speech (TTS), Information Retrieval (IR), shallow parsing, Information Extraction (IE) and semantic role [22] .

Dajaneen/PN kaleessa/AD Finfinne/NN deeme/VB.

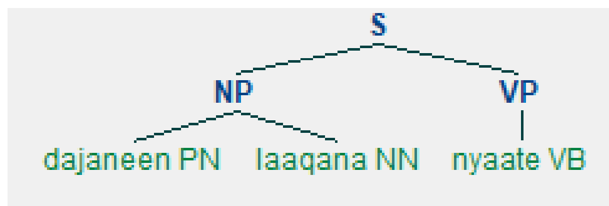
In the above example, words in the sentence, *Dajaneen kaleessa Finfinne deeme,* are tagged with appropriate lexical categories (tag set) of proper noun, adverb, noun and verb respectively. The tag set PN, AD, NN, VB are tags for proper noun, adverb, noun and verb respectively. So the process of tagging takes a sentence as input, assigns a POS tag to the word or to each word in a sentence or in a corpus, and produces the tagged text as output. One of the most popular components of natural language processing systems is the part-of-speech (POS) tagger. It is particularly important for the development of parsers, machine translators, speech recognizers and SRL.

2.3.3.2 Parser

Parser according to a given grammar, parsing (syntax analysis or syntactic analysis) is described as the process of detecting the structure of a certain sentence, such as noun phrase (NP), verb phrases (VP), noun (N), verb (V), and so on [23]. It also deals with a number of sub problems such as identifying constituents that can fit together, testing the compatibility of number and tense [23].

An important task on the road to natural language understanding is parsing of sentences, that has immediate applications in tasks such as phrase recognition, conceptual parsing, machine translation, question answering, spell checker, text summarization, etc. [23]. It is also a crucial application for SRL in order to easily identify all constituents of sentences. Phrase-structure and Dependency-structure parsers are the two categories of existing parsers according to their annotation schema.

Phrase structure: - It focuses on identifying phrases and their recursive structure. X-Bar theory is used to flesh phrase structure rule and it is part of Chomskian linguistics.



X-bar Theory

- ✓ Specifier Rule: XP→(YP) X ‘
- ✓ Modifiers Rule: X ‘→(ZP) X ‘
- ✓ Complement Rule: X ‘→ (WP) X

Terminology

- ✓ Specifier (YP): daughter of XP, sister to X **‘.**
- ✓ Adjunct (ZP): daughter of X **‘**, sister to X **‘.**
- ✓ Complement (WP): daughter of X **‘**, sister to X.

Dependency structure: - It normally be viewed as triples consisting of a head word, relation and a dependent. E.g., in the sentence “*Dajaneen laaqana nyaate “ Dajaneen* ,has a subject relation with the predicate “*nyaate” and “laaqana*” has an object relation with the predicate “*nyaate*” . This relation can be written as triple (*Dajaneen, subject, nyaate*) Dependency tree can be represented with arrows pointing from the head to the dependents or from the dependents to the heads.

Heads: It is the phrasal head of the encapsulating syntactic constituent. For example, the head of a noun phrase node is a noun. Every node has at most one head daughter.

Complements: The way the thematic structure of the head must be interpreted is determined by its complements. Examples of complements are subject, direct object, indirect object, as well as arguments with a less obvious thematic role such as verbal complements. A node can only have one complement daughter of every category, e.g., one subject, one direct object, etc.

Modifiers: Modifiers mark such notions as time, place and quantity. It can be omitted without affecting the thematic structure. A node can have several modifier daughters.

2.3.3.3 Lexical Resource

FrameNet

FrameNet is based on a theory of meaning called Frame Semantics, deriving from the work of [24]. The core premise is that most words' meanings may be best understood by using a semantic frame, which is a description of a sort of event, relation, or entity and the participants in it. It is an electronic resource and a framework used by lexicographers for explicit description of the lexical semantics of words, but also by systems for natural languages processing [16]. The key concept in the FrameNet method of annotation is a semantic frame, that can be described as a representation of an object, event or situation. Each frame has its own set of roles. For example, the roles defined for the frame *research* are field, *question*, *researcher* and *topic*. A frame that are semantically related to the frame are evoked by verbs. For example, the frame research is evoked by *investigation* and *research*. Roles in FrameNet are called frame elements (Fes), the frame-evoking words are called lexical units (Lus) [25].

In Frame Net Each semantic frame is used defined with respect to its frame elements, which are fine-grained semantic role labels. A frame is a structure that defines a word's semantic meaning. It's a generalizable notion with intuitively recognizable repeated frame pieces. The sentences are ordered in a hierarchical manner with each frame referring to a notion dataset from Frame Net and frame elements are the pieces that make up a frame. Frames at the higher level refer to a more generic concept while frames at the lower level refer to more specific concepts [2]. The frame elements for an individual frame are classified in terms of how central they are to a particular frame [26]. Three levels of semantic roles can be distinguished.

Core Elements: Core elements make the frame unique from other frames, it is Conceptually necessary for the frame, roughly similar to syntactically obligatory.

Peripheral Elements: Peripheral Elements providing additional information about the event, such as time and place. Roughly similar to modifiers & it is not central to the frame.

Extra-thematic Elements: It describing the frame with respect to a broader context and it is not specific to the frame and not standard modifiers.

In general, Semantic frames, which are characterized as a schematic depiction of scenarios including multiple participants, propositions, and other conceptual roles, are the focus of FrameNet [5].The project methodology has followed a frame-by-frame approach, which entails first selecting a semantic frame (e.g Commerce), defining the frame and its participants or frame elements (Buyer, Goods, Seller, Money), listing the various lexical predicates that invoke the frame (buy, sell, etc.), finding example sentences of each predicate in a corpus, and annotating each frame element in each sentence. FrameNet does not employ parse trees, annotators highlight the beginning and end points of frame elements in the text, and add a grammatical function tag defining the frame element's syntactic relation to the predicate.

PropBank

PropBank Although "PropBank" refers to a specific corpus developed by Martha Palmer et al., it is a corpus that is annotated with verbal propositions and their argument [9] . The terms PropBank is now being used as a generic noun to refer to any corpus that has been annotated with propositions and their argument. Generally referred to be a resource of semantically annotated sentences. Generally referred to as resource of sentences annotated with semantic roles [22]. It is a corpus of naturally occurring sentences with manually annotated syntactic structure and semantic roles [27].It is built by assigning semantic arguments to constituents of the hand-corrected Treebank parses [23]. Mainly used for developing systems for natural language understanding that depends on semantic parsing, the labels for the semantic roles were attached to the corresponding nodes in the syntactic trees. The argument names and the syntax tree representation for the sentence.

The Proposition Bank is a useful method of semantic representation that involves layering predicate-argument information and semantic role labels, on top of the Penn Treebank's syntactic structures [27]. Since it's difficult to provide a universal collection of semantic or thematic roles that covers all types of predicates, PropBank defines semantic roles verb by verb. The semantic arguments of a single verb are numbered from 0 to 4 [28]. ARG0 is the argument that shows features of an Agent for a given verb, whereas ARG1 is a Patient or Theme. Core arguments are those with labels ranging from ARG0 to ARG4. Those numerical labels denote semantic roles of a broad nature (adjuncts).

Table 1: Basic argument of PROBANK

No	Tag	Description
1	ARG0	Agent, operator (Agent, causer, experiencer)
2	ARG1	Patient, Theme
3	ARG2	Beneficiary/ Experiencer / Instrument /extent
4	ARG3	Starting point / Beneficiary / Instrument / Attribute
5	ARG4	Ending point

These adjuncts (circumstantial objects) are marked as ARG-Ms (modifiers). They can appear in any verb’s frame. There are 12 secondary tags for ARGMs in the Proposition Bank: DIR, LOC, MNR, TMP, EXT, EXP, REC, PRD, PRP, DIS, MOD, NEG,etcs

Table 2: PropBank List of Annotated Adjuncts

ARGM type	**Description**
DIR	Direction
LOC	Location
MNR	Manner
TMP	Time
EXT	Extent
PRP	Purpose
CAU	Cause
NEG	Negation marker

The numbered arguments of PropBank meant to be interpreted in a predicate specific manner whereas the ARGM'S have a global interpretation. The procedure to design PropBank is based on creating frame sets for verbs and then using them as annotation guidelines for the annotation and the PropBank development process is divided into two parts: framing and annotation [29].

Framing

Framing is the process of creating frames files in PropBank development; it is a collection of frames set entries for a verb. Frame sets are used to identify the predicate and its possible arguments. Frame files in PropBank include verb-specific descriptions of all possible semantic roles. Example, frame set for the verb **raggasise** (approved).

Roles

- ✓ ARG0: approver
- ✓ ARG1: thing approved

Manni marichaa labsii lama raggasise (The Council approved two draft edicts)

- ✓ ARG0: Manni marichaa
- ✓ REL: raggasise
- ✓ ARG1: labsii lama

Annotation

Annotation the way to choose appropriate argument role for the given predicate in the sentence. for example, Choosing ARG0 over ARG1 is different b/c

- ✓ The ARG0 label is assigned to arguments which are understood as agents, causers, or experiencers.
- ✓ The patient argument, which is the argument that changes state or is influenced by the action, is frequently referred to as ARG1.
- ✓ ARG0 is the subjects of transitive verbs and a class of intransitive verbs known as unergatives.

Dajaneen laaqana nyaate

- ✓ Dajaneen (**ARG0**) laaqana nyaate
- ✓ Dajaneen (**ARG0**) rafe

Semantically external arguments have what are called Proto-Agent properties such as

- ✓ Volitional involvement in the event or state
- ✓ Causing an event or change of state in another participant
- ✓ Movement relative to the position of another participant

Internal arguments (labeled as ARG1) are the objects of transitive verbs and the subjects of intransitive verbs called unaccusatives:

- ✓ **Tolan fodda** (ARG1) **cabse**
- ✓ **Foddan** (ARG1) **cabe**

These arguments have Proto-Patient properties, which means that these arguments

- ✓ Undergo change of state
- ✓ Are causally affected by another participant
- ✓ Are stationary relative to movement of another participant

So, annotation the way to assigning appropriate role, tag and name for the argument of the predicate, in the sentence with specified rules of PropBank annotation mechanism.

2.4 Approaches to Semantic Role Labeling

In recent years, there has been a lot of interest and engagement in creating novel techniques to learning for NLP [29]. Several learning methods have been used including Rule-based, Statistical or Connectionist and various hybrid approaches. Rule-based approaches such as Head-Driven Phrase Structure Grammar (HDPSG) even they have their own shortcomings, like they are time-consuming and have limited coverage. Most of the SRL approaches are statistical.

2.4.1 Rule-based Approach

This method is based on the explicit description of linguistic facts using well-understood knowledge representation schemes and associated algorithms, and it has been used to construct a number of natural language processing systems [30]. Rule-based approach usually involves of a set of rules, an inference engine, and a workspace or working memory. Knowledge is represented as facts or rules. A rule whose condition is satisfied is repeatedly selects and executes by inference engine. Human-developed rules (e.g grammatical rules) and lexicons are the primary source of evidence in rule-based systems. The rule-based approach has an advantage over the corpus-based approach in the following situations:

- ✓ Under-resourced languages, for which large corpora, possibly parallel or bilingual, with representative structures and entities are neither available nor easily affordable, and
- ✓ To morphologically rich languages, for which data sparsity exists even when corpora are available [30].

2.4.2 Statistical Approaches

Statistical NLP encompasses all quantitative techniques to automated language processing, including probabilistic modeling, information theory, and linear algebra [31]. It frequently employs massive text corpora to create approximation generalized models of linguistic phenomena based on real-world examples provided by the corpora, without requiring extensive linguistic or world expertise. People cannot truly work from seeing a vast amount of language use in its context in the world in statistical NLP. As a result, people merely employ texts and consider the textual context to be a substitute for placing language in a real-world context.

A body of texts is called a corpus. Corpus is simply Latin for "body", several collections of texts formulate a corpus. They are usually characterized by such large text corpora and performing some analysis which uses primarily the text characteristics without adding significant linguistic or world knowledge [29].

2.4.3 Connectionist Approach

Connectionist approaches, like statistical approaches, create generalized models from instances of linguistic occurrences using a network of interconnected simple processing units with knowledge stored in the weights of the connections between units. Connectionist models integrate statistical learning with various theories of representation, which distinguishes them from other statistical approaches. Furthermore, because connectionist architectures are less limited than statistical architectures, language models are more difficult to observe in connectionist systems [29].Learning methods are designed to support automated knowledge acquisition, fault tolerance and plausible induction.

Using learning methods for natural language processing is especially important because learning is an enabling technology for many language-related tasks, such as speech recognition, spoken language understanding, machine translation and information retrieval. Furthermore, learning is important for building more flexible, scalable, adaptable, and portable natural language systems.

The first two approaches rule-based and connectionist are most commonly used by different researchers to develop SRL success has also been achieved by statistical techniques [32].Most of the current statistical approaches to SRL are supervised machine learning technique, requiring large quantities of human annotated data to estimate model parameters. SRL is commonly developed using a supervised learning paradigm.

2.4.4 Supervised Approach to Semantic Role Labeling

The most extensively utilized technique for automatic SRL is supervised machine learning [32]. To establish what counts as a predicate, define the set of roles used in the task, and provide training and test sets, the current approach to semantic role labeling relies on role-annotated data such as FrameNet and PropBank resources [22] .Supervised systems offer a more flexible approach to role labeling. By training them on different sets of data they can be easily adapted to different text domains and even to different languages [22].

Supervised machine learning contains of three major steps. In the first step, the system is trained on the text where segments of texts are already correctly labeled (with semantic roles in this task). It reads the text (training input) and collects the knowledge about the occurrences of labels. In the second step, the system reads a new text for which the labels are to be automatically assigned (test input) and attempts to predict the correct label for every given segment of the text using the information available in the text and the knowledge acquired in training. In the third step, the performance of the system is evaluated [20].

2.4.5 Unsupervised Approach to Semantic Role Labeling

It attempts to leverage regularities in unlabeled data in order to produce predictions without the use of manually annotated training data, i.e., it uses unlabeled data for semantic role labeling and is also known as a rule-based method. It associates instances with class labels as described by the task, such as semantic role labels, but not with class labels as defined by the task.

This link must still be built, either using data from a manually developed lexicon or by manually labeling a small collection of instances. Unsupervised semantic role labeling eliminates the requirement for time-consuming manual text labeling (it does not use role-annotated data) and allows for the usage of a large, representative corpus [33]. Argument classification is a clustering problem that involves assigning argument instances to groupings. Each cluster should preferably include arguments corresponding to a single semantic role, with each role being assigned to only one cluster. Unsupervised systems are very beneficial when no training data is available.

2.4.6 Semi-Supervised Approach to Semantic Role Labeling

A major limiting aspect is the need for vast manually labeled semantic resources to train semantic role labeling algorithms for supervised approaches. Because they leverage supervised learning as sub-routines in the estimate process, existing semi-supervised approaches to SRL can mainly be considered extensions to supervised techniques. In addition to manually annotated corpora, a semi-supervised approach to semantic role labelling requires only a small manually labeled corpus of role-annotated utterances. It also makes use of information from unlabeled corpora [34]. In semi-supervised learning paradigm assumes that manual creation of a small labeled resource of seed annotations is feasible, and that in addition to these labeled instances, there is a large number of unlabeled instances.

2.4.7 Features Used in Assigning of Semantic Roles

Supervised learning algorithms need to be trained before they can be applied to unseen data using on a set of training data, as opposed to the unsupervised methods. The common type of machine learning algorithms applied to the SRL problem is classification algorithms. The classification algorithm goal is to assign class labels to a set of instances automatically. Instances represent the items to be classified and their classes. So, instances are usually encoded as a set of features. Each feature describes a different aspect of the item to be classified. Finding a feature set that yields an accurate classification is an important and certainly not trivial task [10].As researchers have explored new ways the set of features used in SRL systems has grown over time in order to leveraging the syntactic analysis of the entire sentence to better analyse specific semantic roles. Early SRL systems used only a few of features, but current state of the art systems uses dozens. However, the features used in the earliest systems continue to form the core of current SRL.

The following are features that were applied for first time to SRL assignment by [10].

Phrase Type: - Reflects the reality that some semantic roles are best fulfilled by one sort of phrase while others are best fulfilled by another. Different roles tend to be realized by different syntactic categories, for example, Speaker is likely to appear as a noun phrase in FrameNet communication frames, while Topic is likely to appear as a prepositional phrase or noun phrase, and Medium is likely to appear as a prepositional phrase, as in: "[**Speaker** We] talked [**Topic** about the proposal] [**Medium** over the phone]." Phrase types were derived automatically from parse trees generated by the parser.

Governing Category: -Defines the grammatical function of the constituent that realizes a particular semantic role. It is based on the fact that some semantic roles are realized as the subject in a sentence, and others are realized as the direct object. The feature called "governing category", or gov, has only two values, S and VP, corresponding to subjects and objects of verbs, respectively. This feature is restricted to apply only to NPs, as it was found to have little effect on other phrase types. If a constituent bearing a semantic role is governed by the node S in a syntactic tree, it is the subject of the sentence, if it is governed by the node VP; it means that it belongs to the verb phrase, which is the position of the object.

The difference between the direct and the indirect object is not made. This feature restrictedly applied to NPs so if NP nodes found under S nodes are generally grammatical subjects, and if NP nodes under VP nodes are generally objects.

Position: - In order to overcome errors due to incorrect parses, as well as to see how much can be done without parse trees, Gildea and Jurafsky [10] use position as a feature. This feature merely indicates whether the constituent to be labeled appears before or after the semantic frame's predicate. Because subjects come before verbs and objects come after, this trait is significantly associated with grammatical function.

Voice: -The distinction between active and passive verbs plays an important role in the connection between semantic role and grammatical function, since direct objects of active verbs often correspond in semantic role to subjects of passive verbs.

Head Word: - Lexical dependencies are extremely important in labeling semantic roles as indicated by their importance in related tasks such as parsing. Head words of noun phrases can be used to express selection restrictions on the semantic types of role fillers.

Sub-Categorization: This feature refers to the set of a verb's syntactic arguments in the sentence. For example, we expect an intransitive use of close such as the door closed to have a different mapping from syntactic to semantic roles when compared to the transitive use in "He closed the door". The sub categorization feature of the first verb usage is {subject}, while the sub categorization for the second is {subject, object}.

Argument Set: This feature, called Frame Element Group in the FrameNet-based system of Gildea and Jurafsky [10] is the set of all roles appearing for a verb in a given sentence. Since this feature depends on the roles assigned to all constituents in a sentence, it was employed in a post-processing ranking of role assignments for an entire sentence.

The following are features introduced by Pradhan [28]

Argument Order: This feature is an integer indicating the position of a constituent in the sequence of arguments for the given verb. It is computed after an initial phase classifies constituents as arguments or non-arguments. Because the feature does not use the syntactic parse tree, it can help make a semantic role labeling system robust to parser error.

Previous Role: This feature is simply the label assigned by the system to the previous argument (skipping non-argument constituents), because this feature introduces a dependency among labels assigned to different constituents.

Head Word Part of Speech: Because Penn Treebank part of speech categories distinguish singular from plural nouns, and proper and common nouns, the part of speech tags of the head of an NP can refine the type of noun phrase involved.

Named Entities in Constituents: This feature uses the named entity recognizer to identify words as instances of the classes Person, Organization, Location, Percent, Money, Time, and Date. This feature helps handle the data sparsity caused by the unlimited sets of proper names for people, organizations, and locations in particular.

Verb Clustering: Because semantically similar verbs such as eat and devour will occur with the same objects, they will be assigned to the same clusters. The use of this cluster for predicting argument structure is motivated by the observation that semantically similar verbs undergo the same pattern of argument alternation.

Head Word of Objects of PPS: When a verb's argument is a prepositional phrase (PP), the PP‟s head word is the preposition. While this can often be a reliable indicator of semantic role (for example in, across, and toward usually indicate location), most prepositions can be used in many different ways, and the sense can be determined by the preposition's object. For example, in February indicates time, while in New York indicates location.

First/Last Word/POS in Constituent: As with the previous feature, this feature provides more specific information about the argument filler than the headword alone, but does so in a more general way that is robust to parser error and applies to any type of constituent.

Constituent Order: This feature is related to the argument order feature above but is designed to be used in discriminating arguments from non-arguments. The position of each constituent is calculated relative to the predicate, which helps favor constituents close to the predicate.

Constituent Tree Distance: This feature has the same motivation as the previous feature but takes syntactic structure into account.

Constituent Context Features: These features provide information about the parent and left and right siblings of a constituent. For each of these three constituents, the phrase type, head word, and the head word's part of speech are given as features, for a total of nine features.

Temporal Cue Words: A number of words which indicate time, but which are not considered named entities by the named entity tagger, are included as binary features.

Deep Learning

In recent years, deep learning has become not only an emerging machine learning technique but a hot topic in the artificial intelligence field. Deep learning applies multiple layers of artificial neural networks (ANN) to learn representations of data on different levels of abstraction. This method has significantly improved the performance on various machine learning tasks such as object detection, natural language understanding, speech recognition and many other research topics [35].

Conventional machine learning methods usually require considerable work on feature engineering before training. This pre-processing demands much expert knowledge and experience. Unlike conventional methods, deep learning techniques learn multiple representations out of each layer from raw data. The features of raw data are automatically discovered by each processing layer. Therefore, the most attractive feature of deep learning is that it has a general-purpose learning ability, by which a human does not need to design feature templates. An artificial neural network is a mathematical model inspired by biological observations. It consists of a set of connected units named artificial neurons, in a way like the nature of animal brains [36].

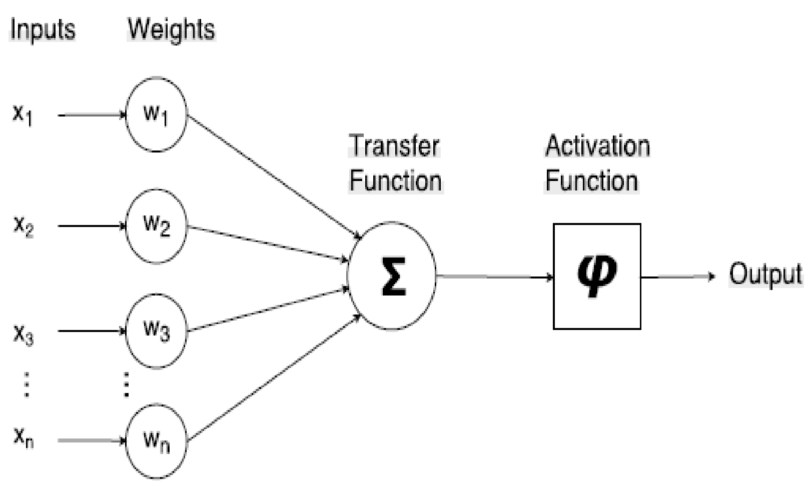


Figure 1: Simple artificial neuron mode (ANN)

Signals can flow through the connections between two neurons. One neuron will receive information and then passes to the next neuron connected to it after processing. There can be weights on the neurons and connections.

The weight value indicates the strength of the signal that it sends to the next one. Neurons may have activation functions inside them, which process the signals coming in. Data flows in the network in a defined direction from input to output. Developers can finally obtain the result produced by the network model from the output layer.

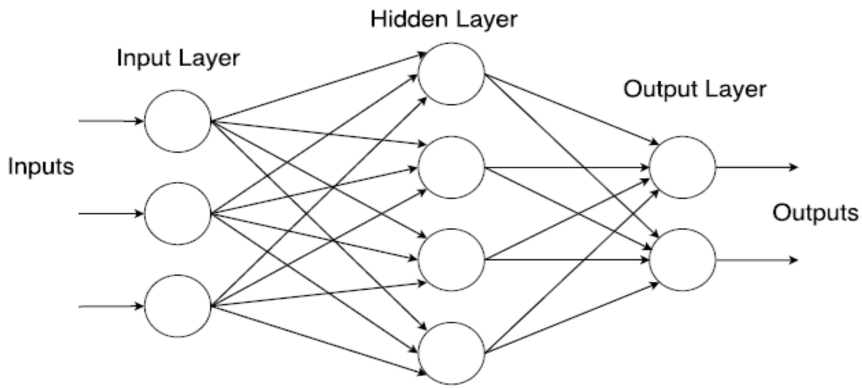


Figure 2: A simple feed-forward neural network model

Training an Artificial Neural Network

The way to make a neural network work for a specific task is supervised learning. Assume that we would build a classification system using the neural network. We may first collect a dataset as the training set, in which each entry is labeled with a category. The network will be trained by feeding the data entries while reading the labeled targets for each entry. The goal of training is to make the output gain the highest possibility value on the desired category, but it is not likely to achieve that at the beginning.

What we need to do is to minimize the disparity between the output and the ideally correct label. In order to achieve this objective, we first define an error function to measure the disparity and then modify the parameters of the network to reduce this error function during training. Even a simple feed-forward neural network may have numerous adjustable weights, and a typical deep learning system may contain millions of these parameters. In general, the error function obtained as the learning process goes on is a multi-variable function, and the work of finding the local minimum is an optimization problem.

Because finding the global optimum is challenging, the learning algorithm's goal is to discover a local minimum. The learning algorithm computes the gradient of the error function with respect to each weight parameter in the high-dimensional parameter space to make the gradient climb downward. We use optimizer iteratively during training to adjust the parameters of the neural network, and finally find a local minimum of the error function. The gradient of the objective function with respect to the parameters Θ is updated through iterations:

$$\Theta = \Theta - \alpha . \Delta_{\Theta} J(\Theta)$$

Here the parameter Θ is called ’learning rate’, which is used to determine the speed of updating parameters. In real world machine learning problems, variants of gradient descent are applied based on how much data we use. Batch gradient descent (BGD) takes the entire dataset to try one update. This method works inefficiently when the dataset is large, and it might be impossible to fit in the memory of one computer. Another method is called Stochastically Gradient Descent (SGD). This method updates the parameters on one pair of training sample, the input x and the target y once at a time.

In order to perform SGD on a multilayer neural network, we first need to obtain the gradient. Backpropagation is one algorithm to calculate the gradient of an error function with respect to the weight parameters of a deep model [32]. The idea simply comes from the ’chain rule’ for derivatives the gradient of the error function at the output with respect to the input of a network can be obtained by tracking derivatives reversely, that is, it can be computed backward from the gradient of the output of this network. By performing backpropagation iteratively from output to input, layer to layer, we can finally have the gradients with respect to the weights of the whole network.

2.4.8 Recurrent Neural Networks

A recurrent neural network (RNN) is a sort of artificial neural network model in which units are linked in a directed cycle. In jobs requiring sequential input, such as speech recognition and machine translation, RNN perform much better [36].

This network architecture can process the input sequence one element at one time step. It has an internal structure which is called 'state' to memorize the information from all the previous inputs passed through.

$$RNN\ (\mathbf{s0};\ \mathbf{x1}:\ \mathbf{n}) = \mathbf{s1}:\ \mathbf{n};\ \mathbf{y1}:\ \mathbf{n}$$

$$\mathbf{si} = R(\mathbf{si-1}, \mathbf{xi})$$
$$\mathbf{yi} = O(\mathbf{si})$$

Here **si** indicates the state at time step **i**, and **xi, yi** represent the input and the output at time step **i** respectively. **R** is a recursively defined function while **O** is a function that maps the state **si** to the output **yi.** A recurrent neural network topology is usually constructed by ***RNN*** modules concatenated with each other. The structure can be described as follow:

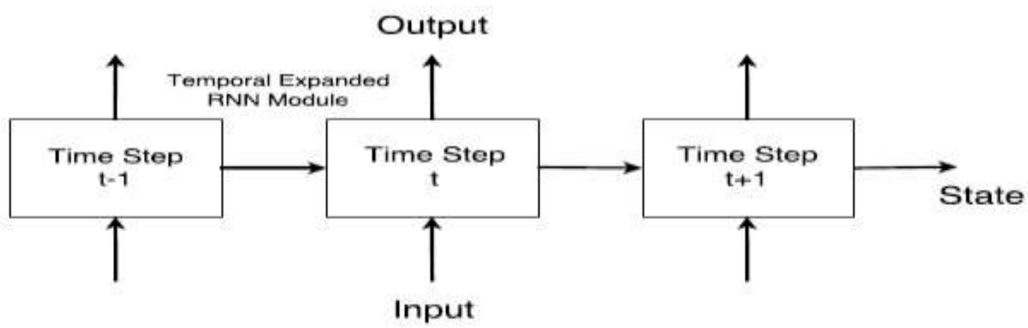


Figure 3: Chain of a RNN module

2.4.9 Long short-term memory (LSTM)

Sepp Hoch Reiter and Jürgen Schmid Huber in 1997 [37] invented kind of recurrent neural network architecture Called Long short-term memory (LSTM). On LSTM there is more improvements were carried out, it has become a rather useful model in research fields such as language processing and speech recognition. Since the conventional Back Propagation Through Time (BPTT) has drawbacks that the gradients of error may either explode or vanish, so the LSTM architecture was proposed. Thus, the error backflow problems in LSTM have had an efficient solution. LSTM can learn and remember not only information for the short term but for long periods of time while a conventional RNN does not have the ability to learn it.

That is, LSTM is an effective solution for the long-term dependency problem [36].The key insight of LSTM is to memorize the information from the previous input and recall it afterward when needed. LSTM is also a kind of RNN module, but the structure is different from an original one. In the figure below we can see the structure of an LSTM cell. An LSTM memory block contains three gates as the figure describes: the input gate, the forget gate and the output gate. The output of each gate depends on the current input **xt** and the previous state **ht−1**. When we walk through each gate mathematically [36], for the input gate**,** a sigmoid layer decides which value should be updated.

$$i = \delta\ (xtW^{xi} + ht\text{-}1W^{hi})$$

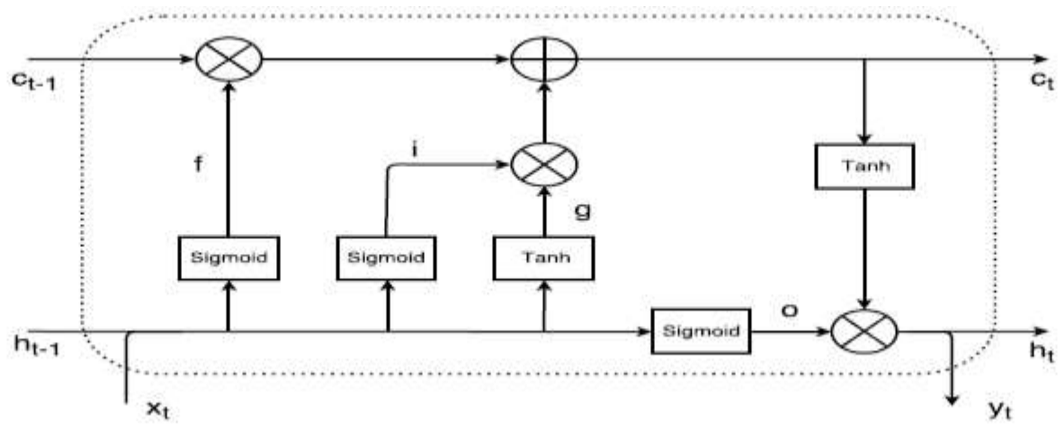


Figure 4: Illustration of an LSTM cell

Meanwhile, a candidate vector ***g*** used to update the state is calculated. It takes ***xt*** *and* ***ht−1*** passing through a **tanh** function. $g = \tanh\ (xtWxg + ht\text{-}1Whg)$

In the forget gate, a sigmoid function takes in ***xt*** and ***ht−1*** to decide how much information should be kept or dropped. $f = \delta\ (xtWxf + ht\text{-}1Whf)$

Then the cell state is updated to ***ct***. Here the LSTM unit carries out both the forget operation and the update operation and then achieves a new state: - $ct = ct{-}1\ f + gi$

The output gate uses a sigmoid function: - $o = \delta\ (xtWxi + ht{-}1Who)$

Finally, the ***ct*** passes through a ***tanh*** function and is multiplied by ***o*** element-wise to derive the output **ht:** **ht = tanh(ct) δo**, this is also the output of the LSTM unit: **yt = ht.**

2.5 Related work

2.5.1 Semantic Role Labeling using Memory Based Learning

The first Persian corpus based SRL system using memory-based learning model and standard features proposed by [39]. They draw the input data from Hamshahri corpus. It was hand-labeled with required syntactic and semantic information. They proposed Twelve semantic roles in the study which were divided into two classes: primary and general roles.

- ✓ Primary Roles: are predicate specific such as: Agent, Patient, Source, Goal, Topic, Percept, Instrument and Beneficiary.
- ✓ General Roles: are assumed to apply across all verbs. They were optional for an event but supplied more information about an event including: Location, Time, Manner and Reason.

The authors implemented SRL in two phase architectures. First, the arguments were identified using a shallow syntactic parser or chunker and second, labeled them with appropriate semantic role with respect to the predicate of the sentence using a machine learning method to distinguish different roles such as Agent, Goal, etc. Both phases were treated as a multi-class classification problem. The classifier was trained in a supervised manner from human-annotated data and using memory-based learning. Some syntactic properties of arguments were used as feature set. The authors mentioned that, the feature set plays an important role in MBL performance and mainly they selected features from the review of previous literature, investigated each of these features in Persian, some acted quite similarly to English, while others showed interesting differences.

Current argument phrase type (the syntactic type of constituents (NP,PP,VP,ADV,SP), Previous argument phrase type, Next argument phrase type, Position and Verb Class (These classes are based on the semantic roles each verb can take) features showed interesting patterns. TiMBL software were used for carried out the experiments. Regarding the learning algorithm, the IB1 classifier was used and parameterized by using overlap as the similarity metric, information gain for feature weighting, using k-nearest neighbors, and weighting the class vote of neighbors as a function of their inverse linear distance.

For the evaluation of their system, they used three measures: precision, recall and F-Score (a combined measure). Using Gold-standard parses, the results showed an F-score of 90.3 % on the argument boundary detection task and an F-score of 87.4 percent on the semantic role labeling task. On a full SRL system, the total system performance exhibited an F-score of 83.8 %.

The objective of an approach to automatic semantic role labeling in Dutch corpora presented by [4] was to annotate a small portion of the D-Coi corpus with semantic roles automatically by using TiMBL, a memory-based classifier. A novel Alpino dependency tree-based technique to rule-based tagging has been developed. Because there was no semantically annotated Dutch corpus that could be utilized as training data, Phrase type (category label), dependency label, POS-tag, position, argument head-word, Head-word POS tag, and dependency label features were employed to describe the candidate argument.

MBL classifier was employed, for the task of automatic semantic role labeling. The evaluation result showed that the TiMBL performance on training data using the LOO method was 65.05% precision, 65.53% recall and 65.29% F-score.The best performing system reached F score of 80%. The system would rank in the lower regions, this is due to the following reasons:

- ✓ The candidate argument selection method used was not perfect. The author did not perform any pre-post-processing steps.
- ✓ The author did not use a very basic set of features.
- ✓ The size of training set was small and they didn’t attempt to optimize TiMBL parameters

Eskedar Yirga [40]proposed the first Amharic SRL using memory-based learning model and Amharic features without semantically annotated corpus of Amharic language. In their study, the assignment of semantic roles to Amharic sentences was discussed. With a modest 551 instance training set, good semantic parsing results for Amharic can be attained in order to evaluate the performance of the model they conducted the experiment with 240 sentences which helps to show different verb class. From these sentences 551 instances were extracted and 21 different classes occur. They are used all available data except one example as training material based on leave one out cross validation (LOO-CV) method.

First Memory based learning can successfully be applied to assign semantic role for Amharic. Second, basic syntactic and lexical features are useful for semantic role labeling. As well as attempting optimize TiMBL parameters significantly improve the system performance. The authors used a MBL tool called Tilburg Memory-based Learner (TiMBL) for training of the classifier and testing of the classifier and their experiments were carried out with the TiMBL software. Regarding the learning algorithm, the IB1 classifier is used, parameterized by default and using MVDM as the similarity metric, information gain for feature weighting, using 3k-nearest neighbors, and final decision as a function of their distance.

The authors investigated different features that were proposed for another language in Amharic, finding that some behaved similarly to English while others displayed interesting differences. They were selected six features that have good discriminative power for Amharic for their study. They clustered the selected features for the study into three. The first cluster characterizes target predicate, the second cluster characterizes the candidate argument and the last cluster characterizes relationship between candidate argument and predicate.

The phrase type, the target verb, the constituent's syntactic category (NP, PP, etc.), and the target verb's subcategorization all worked similarly to English features. In their work, five features (path, position, term category, head word, and voice) revealed fascinating patterns. Their system achieved 69.8% precision, 50.7% recall and 58.3% F-score for unseen verbs. The authors used four measures for the evaluation of their proposed system: accuracy, precision, recall and F-Score (a combined measure). The evaluation result showed that the accuracy in classifying semantic role achieved 82.51% accuracy and 78.19% F-Score with default parameter and 89.29% accuracy and 79.77% F-Score with optimized parameter setting for semantic role labeling task.

2.5.2 Semantic Role Labeling Using Deep Learning

Wang [41]proposed a new method without using syntactic parsing and POS technologies for labeling Chinese with semantic roles. Clustering and labeling were the two subtasks he used. Clustering was chosen Partially to replacing syntactic parsing, during which similar sentences are clustered together. In the labeling step, artificial neural networks were created as many as the number of clusters, each of which used to summing up features of chunks of a sentence and then labeling them with semantic roles.

The researcher hypothesizes that particles and verb meaning were closely related to semantic roles arguments. Particles and verbs meaning were the most important features used in clustering and in identifying semantic roles. In their experiment, more than one thousand Chinese clausal sentences are annotated manually, in which assertions, imperatives, queries are all included. All the syntactic elements are sorted into four categories: argument, predicate verb, particle, and marker. As a result of effective clustering the experiment result shows this method is useful and 83.8% correctness on average was accomplished. Clustering makes the mess of sentences display orderly so that some useful features can be summed up.
However, when the size of clusters is considered, the system performs unsteadily, because shorter sentences tend to be clustered easily that noise in the clusters hindered artificial neural networks to learn useful knowledge.

End-to-end semantic role labeler for the Portuguese language that address the SRL problem as a supervised sequence labeling task done by [42]. They used one-step the BIO tagging schema and a word embedding model in a deep bidirectional long short-term memory neural network (deep BiLSTM). In their work the network predictions are inputs to an inference mechanism that uses a global recursive neural parsing algorithm, specifically tailored for the SRL task. The method they used requires a minimal feature engineering process and does not depend on syntactic parsing. In addition, the authors provide a throughout investigation on the effects of word embedding dimensionality and network depth on the overall performance of their approach. In their experimental results with the PropBank corpus SRL approach outperforms the state-of-the-art approach reported on literature for the Portuguese language.

The author’s employ the skip-gram model application on the full dump of the Brazilian Portuguese version of Wikipedia corpus to word representations. In their architecture, given a sliding window of words, one attempts to predict the adjacent words (the context) based on the central word (the target token). The researcher hypothesizes that the model offers good representation for rarely seen tokens and outperformed other models in NLP tasks such as sentiment analysis and syntactic parsing.

The authors, were used the NLTK Punkt tokenizer for sentence splitting on each of its articles. Sentences obtained were then lowercased after extracting the raw text from the Wikipedia corpus, followed by a series of transformations that included accents removal, punctuation separation, and substitutions. At last, they tokenized each resulting sentence and feeding the skip-gram training algorithm. They trained three distinct word representations with 50, 100 and 150 dimensions, respectively. [42] suggest that a vector dimensionality between 50 and 150 yields the best accuracy values for extrinsic tasks and it is worthwhile to carefully choose word embedding dimensionality for such tasks. In all three models, they employed a context-window of size 5, discarding the tokens with a total frequency lower than 5 and train their model for ten iterations with an initial learning rate of 0.025 that linearly decays until reaching the minimum learning rate of 0.0001.

They also conducted an extensive investigation about the effects of two crucial factors on their structure, the depth of network architecture and the proper word embedding dimensionality. Their approach outperformed the previous state-of-the-art on the Portuguese language by 3.05 F1-score points, reducing the relative error in 8.74%. they also confirmed the hypothesis that picking the optimal embedding dimensionality is critical for obtaining the best accuracy on SRL task. their final model was based on word vectors with 150 dimensions passing through a deep network with two BiLSTM layers.

Emma et al [43] present linguistically-informed self-attention (LISA), a neural network model that combines multi-head self-attention with multi-task learning across dependency parsing, part-of speech tagging, predicate detection and SRL .Unlike previous models which require significant pre-processing to prepare linguistic features, LISA can incorporate syntax using merely raw tokens as input, encoding the sequence only once to simultaneously perform parsing, predicate detection and role labeling for all predicates. Syntax is incorporated by training one attention head to attend to syntactic parents for each token. Moreover, if a high-quality syntactic parse is already available, it can be beneficially injected at test time without re-training our SRL model.

The authors experiment with both standard pre-trained GloVe word embeddings [44] and pre-trained ELMo representations with fine-tuned task-specific parameters [45]in order to best compare to prior work. They select Hyperparameters that resulted in the best performance on the validation via a small grid search, and they trained their models for a maximum of 4 days on one GPU using early stopping on the validation set. They convert constituencies to dependencies using the Stanford head rules v3.5 [46].

In their experiments on CoNLL-2005 SRL, LISA achieves new state-of-the-art performance for a model using predicted predicates and standard word embeddings, attaining 2.5 F1 absolute higher than the previous state-of-the-art on newswire and more than 3.5 F1 on out-of- domain data, nearly 10% reduction in error. On ConLL-2012 English SRL their model also shows an improvement of more than 2.5 F1. Their model out-performs the state-of-the-art with contextually-encoded (ELMo) word representations, by nearly 1.0 F1 on news and more than 2.0 F1 on out-of-domain text. Finally, they conclude with statement of LISA out-performs the state-of the-art on two benchmark SRL datasets, including out-of-domain

Table 3: Summery of related work

Language	Approach	Feature used	Performance
Persian	MBL	- Current phrase type - Previous argument phrase type - Next argument phrase type - Position – Verb Class	87.4% F-score
Dutch	MBL	- Phrase type - Dependency label – POS-tag - Position- Argument head-word - Head-word POS tag - Category label and dependency label	65.05% precision, 65.53% recall And 65.29% F-score
Chinese	ANN	- Position – Verb meaning - Markers<0>.	83.8 % accuracy
Amharic	MBL	-Term category, voice, - phrase type, syntax, - verb class and head word	82.51 % accuracy and 78.19 % F-Score
Portuguese	RNN	-Word embedding vectors	67.62% Precision, 68.75 % recall 68.18 % F1-score
English	Deep learning LISA	-Pretrained word embedding	98.9 %, Precision 97.9 % Recall 98.4 % F-score

CHAPTER THREE

3 DESIGNS OF SRL FOR AFAAN OROMO SIMPLE SENTENCE

3.1 Introduction

In the previous chapter some of literatures related to semantic role labeling were described. Review of related works on the area was also discussed. In this chapter, we discuss about Afaan Oromo sentence formation, preprocessing for SRL, component of SRL, stage of SRL, the overall design of the proposed semantic role labeling for Afaan Oromo simple sentence. First, we discuss about component of AOSRL, demonstrate the generalized architecture of the proposed model, and then we describe main component of the proposed model such us the text pre-processing, the training and the semantic role labeling components along with their sub-components for Afaan Oromo simple sentence.

3.2 Research Methodology

This study employs design science approach. Six basic steps are included in the approaches. The steps involved are problem identification, objective definitions, design and development, demonstration, evaluation, and communication. A precise procedure for designing objects to solve observable problems, making research contributions, evaluating the designs, and communicating the results to the right audiences is all part of the design science approach. For better understanding of the work and the approaches that are available, techniques and algorithms used and also to find the gaps from previous works, literature review is used. So, it helps to develop technology-based solutions to important and relevant business problems. To develop and design the model, the researcher may use different ways and techniques that are relating to Afaan Oromo semantic analysis according to the features of the language that are related to semantics.

3.3 Literature Review

Detail review and assessment would be conducted through works which have been done on the area of semantic role labeling technique and related issues. In addition, areas including syntactic analysis, semantic analysis, challenge, approach, features, common list, and type of SRL reviewed since they are building blocks of a natural language. Moreover, different researches conducted on the development of SRL for different language have been examined in chapter two.

3.4 Data Collection

There are no ready-made annotated corpus texts for semantic role labeling or other NLP applications in the Afaan Oromo language. In the corpus, we have gathered the data from various sources of Afaan Oromo text. Those sources are considered to be under various domain or categories such as Afaan Oromo previous research corpus. Accordingly, official Facebook page of (OBN, OMN, FNBC, EBC, BBC Afaan Oromo, Voice of America (Afaan Oromo service)), and websites of (www.fanabc.com) Afaan Oromo section is some of the main data sources.

Corpora can be classified into two types, annotated corpora and unannotated corpora. Unannotated corpora are simply large collection of raw text, whereas annotated corpora are a large collection of text which contains text with their additional information such as transcription, part-of-speech tags, parse trees, etc. In this study we have used annotated data that area annotated with semantic role by the help of scholars around the domain (i.e., Afaan Oromo language). Since we were used simple sentence, first that sentence was collected manually for source we mentioned above, then tagged with semantic role label tag set by the help of linguistics and researcher.

3.5 Architecture of the System

After a review of researches done on Afaan Oromo area, there is no work related to semantic role labeling done before, so there is one problem how identified and select features. Our proposed approach aims at automating this process of feature design by automatically learning features from given labeled data. After learning word representations from labeled data, generated word feature vectors by keras embedding layers are used for training. In this section we will describe an architecture proposed for AOSRL.

This architecture can be used with different deep learning algorithms to use word vectors as a feature for AOSRL.The architecture proposed contains four main processes. These are preprocessing, model building, model training, and finally semantic role labeling with trained prediction model to label the constituent of the sentence or model evaluation. The processes are brief described in next section's:

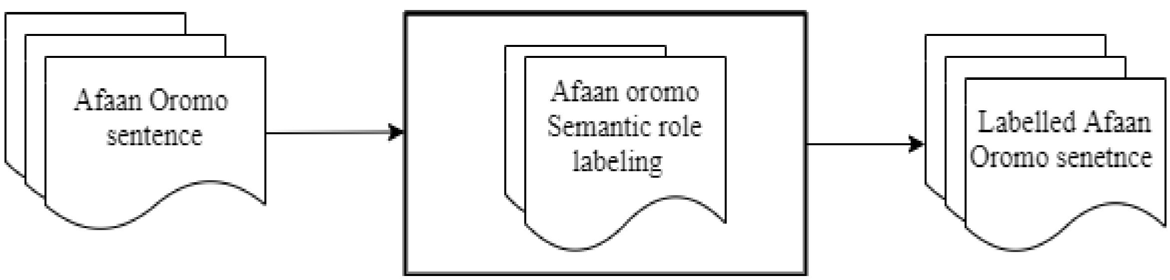


Figure 5: Simple architecture of AOSRL

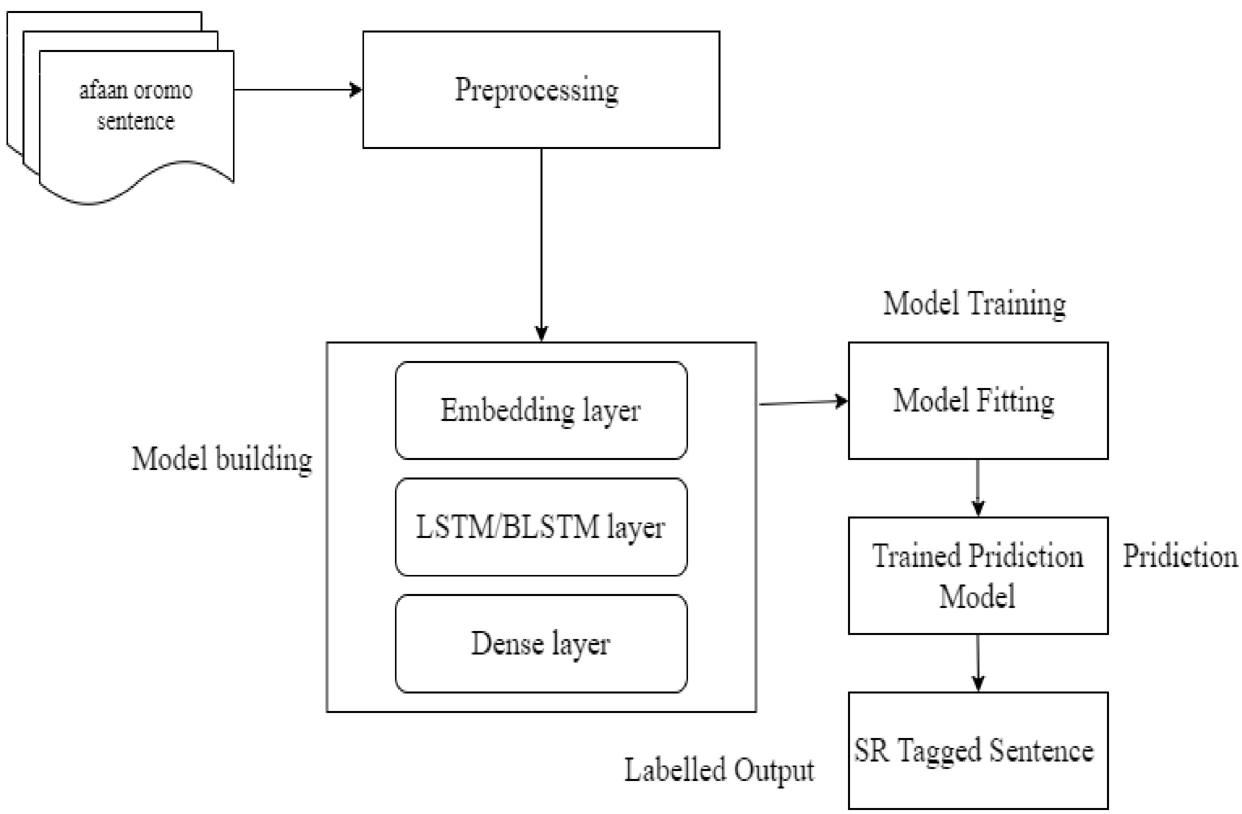


Figure 6: Architecture of AOSRL with each specific component

3.5.1 Pre-processing

Preprocessing stage is the first process in our proposed approach. we are used labeled data for training and building AOSRL model. Before proceeding to next stages these corpora have to pass through two main preprocessing stages. The first stage is cleaning, tokenization and normalization process which replace nan value with fill forward method, converting Afaan Oromo sentence to token(word) and make every token with same case format. After this step is completed the next stage in preprocessing is semantically role labeled is represent in feature format for the model to understand the features of the labeled sentence and to embed those features.

E.g. Simple Afaan Oromo sentence, Konkolaatan **saree ajjese.**

First this sentence is tagged/labeled with semantic role label tag set, by reviewing some of the previous works about SRL we have seen different tag set & investigated some of the tag set from PropBank [9] for Afaan Oromo and we have used BIO tagging style.

snum	word	role
1	konkolaataan	B-ARG0-AGT
1	saree	B-ARG1-PAT
1	ajjeese	VB
1	.	O

Figure 7: Sample BIO tagging

From the above figure, the first column is the sentence number used to group sentence. The second column is the word(token) from the sentence. The third column is semantic role, it annotated using BIO tagging style, which is beginning, inside outside of the phrase of the given sentence and the argument is based on PropBank annotation mechanism as described in Appendix B**.**

In a real-world sequential machine learning task, the entries of a dataset can be of arbitrary length. In our data set the number of tokens in one sentence varies between sentences. Since our model implementation can only take in data entries in a fixed length, we need to either cut up or fill up each sentence in the dataset with some dummy element to an identical length by using some techniques. This pre-processing procedure is often named ’padding’. In this work, we adjust all of the sentences to the length of the longest sentence (the sentence with the largest number of tokens) by using keras preprocessing method pad sequences. Sentences that are shorter than this length will be filled with zero(s) in the gap and from keras library we used to_categorical method to convert the categorical data and feed the value of index to the embedding layer.

3.5.2 Model building

We construct the model from four main of neural network layers. Those network layers are: input layer contain maximum length of sequence, Embedding layer with 100 unit for both LSTM and BLSTM, one hidden layer with 100 input units, dense layer with 100 unit with the help of relu activation function, finally Crf layer for output. Our models were trained using Adam, an efficient adaptative algorithm for gradient-based optimization of stochastic objective functions that is typically suited for high-dimensional parameter [47] and as loss function Crf loss function is used. To prevent overfitting and to improve the overall performance of our model, we used the Dropout technique introduced by [48].The following table (table 6) shows the hyper parameter used in our training model.

1. Embedding layer:

Since computers are not able to read human language, we need to represent words in a way which computers can process, such as numerical vectors. The conventional method is to use a one-hot word representation to encode words into numbers. This method regards each word as an atom symbol and represent each word as one dimension in a vector whose length is the same as the whole vocabulary table. However, this one-hot representation is sparse and cannot reflect any lexical relationship between words. Meanwhile, the dimension of the word vector would become extremely large as the vocabulary table of the corpus grows. Thus, this high-dimensional and sparse representation method does not perform well in a semantic aspect, and it may also lead to computational problems when the dimension goes high (especially for the use of training neural network models). In recent years, *Word2vec* invented by [49] and *GloVe* put forward by Stanford University [50] have become popular methods for producing vector representations for words. These methods are trained to learn a model of lexical semantics from text. They provide not only vector space representations of words but also semantic correlations between words to some degree. Moreover, they also represent the words in a set of low-dimensional dense vectors, which reduces the computational complexity.

There are mainly two model architectures for creating vector representations utilized by Word2vec: Skip-gram Model and Continuous Bag-of-Words Model (CBOW). [51]Introduced a Skip-gram model for learning high-quality vector representations of words. This model demands much lower computational cost than previous neural network architectures because it does not involve dense matrix multiplications. Meanwhile, although the Skip-gram model takes only unstructured data without syntactic information, the word vector representations it learns can have interesting linguistic regularities. The Skip-gram Model uses a word to predict the **N** words appearing before and after it. In other words, the idea is to predict the surrounding words in a window of size **N** in the context. Embedding layer is one of the available layers in Keras**,** in our model we used it to feature generation by taking preprocessed text in the first step as an input. So, word embedding is developed as part of deep learning in Keras that generates a random value to first and iteratively learns the representation of all word.

This stage is where the words semantic and syntactic relations are learned and we initialized this layer with vocabulary size, maximum sequence length and embedding size as we put in table 6. The output from this stage is fixed sized vector representation for each word. After the training process is finished, all the words with their corresponding features will be logged to LSTM layer. So, this layer was used to translate (embed) the dataset to the corresponding vector representations and this vector representation fed to the LSTM and BLSTM layer.

2. LSTM layer: That is, LSTM is an effective solution for the long-term dependency problem [36].The key insight of LSTM is to memorize the information from the previous input and recall it afterward when needed. They were introduced to avoid the long-term dependency problem. In regular RNN, the problem frequently occurs when connecting previous information to new information. If RNN could do this, they'd be very useful. This problem is called long-term dependency. We have used two LSTM layer with 100 unit each and with 0.1 recurrent dropout for regularization method for recurrent neural networks. Then the outputs from the LSTMs are feed to dense layer. Finally, we deploy a CRF layer on top of the network for final prediction, and it gives a probability distribution over possible output label. Figure below illustrates a detailed topology of the deep LSTM neural network part by CRF on top of LSTM for output.

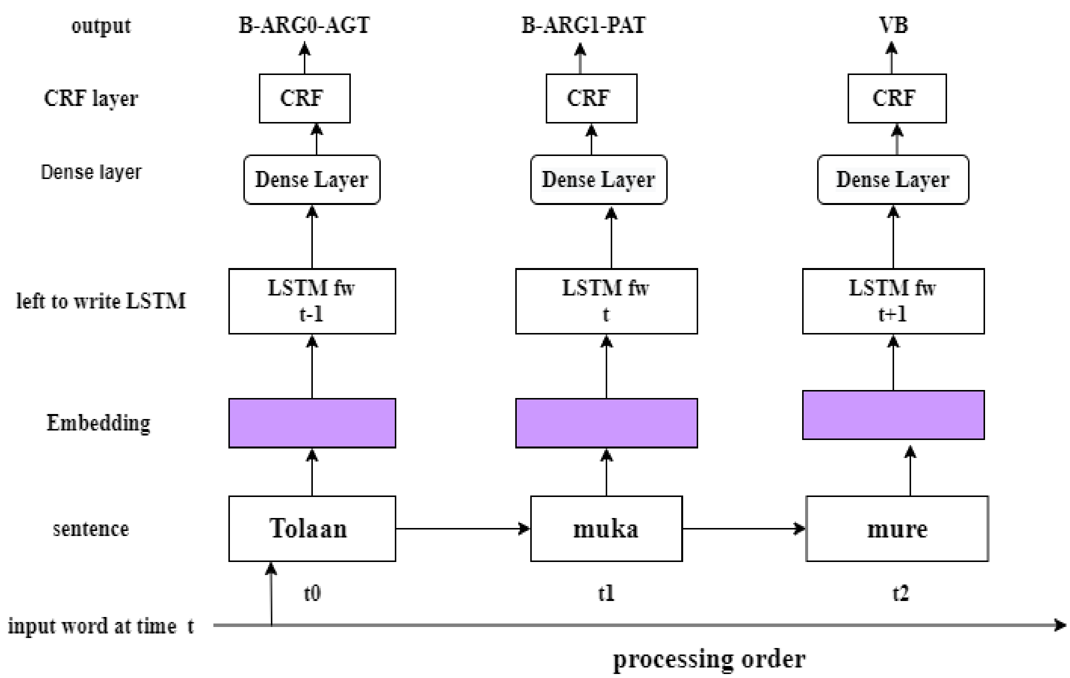


Figure 8: A detailed LSTM network topology

3. Bi-directional long short-term memory (B-LSTM) layer: Bidirectional LSTM (BiLSTM) architecture, in contrast, considers both historical and future steps in order to learn information from preceding as well as future input events. The Bidirectional LSTM (BiLSTM) architecture contains forward (left-to-right) and backward (right-to-left) LSTM layers whose outputs are merged by concatenation in a new layer that, intuitively, encodes past and future information. This arrangement, as occur in other types of multi-layer networks, enables capturing higher levels of abstraction yielding superior performance in sequence labeling tasks such as part of speech tagging, chunking, and named entity recognition.

According to prior experience, adding more layers on the model may achieve better results in a deep learning task However, adding simply the number of layers to make deep it may make the mode performance got a little bit worse. This result indicates that adding more layers have no contribution to a better result in this task, going deeper does not mean going better. The reason behind this fact may be that the deeper model takes more detailed information from the training data, the data we have is small 1800 sentence only, which means it overfits more on the noise from the training data and the model see the training data many times. Another possible reason is that the gradient is vanishing through the backpropagation in a deeper network. This results in a poor prediction on the development data.

So, in our study we used two layers of BLSTM with 100 unit each and 0.1 recurrent dropout values to randomly sets input units to 0 with a frequency of rate at each step during training time, which helps prevent overfitting. The return sequence value is set to true, because we need the output as sequence like in BIO tagging way. We first use a standard fully connected layer with a relu activation function in dense layer. Next, we deploy one LSTM in the forward direction (positive time direction) and one LSTM in the backward direction (negative time direction) taking in the output of the fully connected layer independently. Then the two outputs from the two LSTMs are concatenated together as one vector for each time step. Finally, we deploy a CRF layer on top of the network for final prediction, and it gives a probability distribution over possible output label. Figure below illustrates a detailed topology of the deep BLSTM neural network part by CRF on top of BLSTM for output.

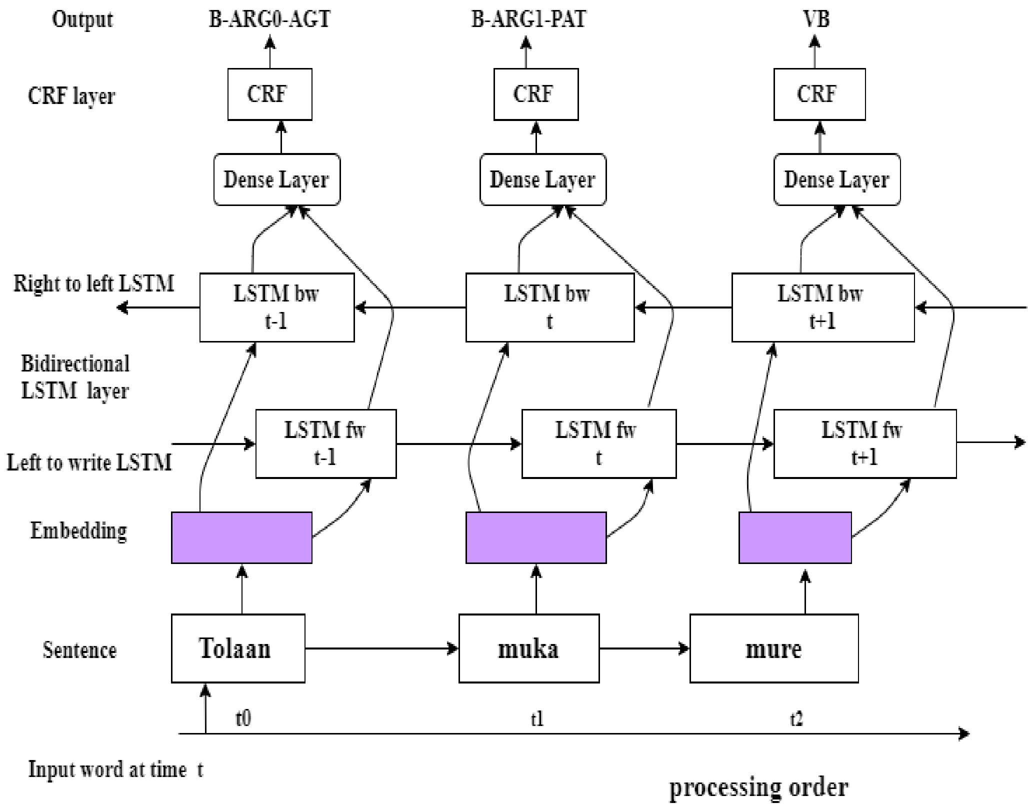


Figure 9: A detailed BLSTM network topology

4. Dense layer: Dense Layer is a Neural Network that has deep connection, which means each neuron unit in dense layer receives input from all neurons of its previous layer and we used dense layer to minimize the dimensionality of the output value. It performs a matrix vector multiplication, and the values used in the matrix are parameters that can be trained and updated with the help of backpropagation. We initialize with number of output label and relu activation function and feed the output of the dense layer to the CRF layer for final output. The CRF layer processes the input with a trained probability transformation matrix. After the calculation, the labels that have the maximal probability values are the outputs. Lafferty, McCallum and Pereira [52] proposed a class of probabilistic models for segmenting and labeling sequential data, the conditional random field (CRF).

5. CRF layer

we deploy a CRF layer on top of the network for final prediction, and it gives a probability distribution over possible output label. CRF models take into account the association between neighbourhood tags and decode them all together, so they are widely used in various sequence labeling tasks such as POS tagging for NLP and gene prediction for computational biology. The CRF is employed on top of a LSTM and Bidirectional LSTM layer [53] [54]and on top of a CNN layer [55] in deep neural network-based NLP application.

For an input sentence S = (s1, s2…), we use N to represent the output features of LSTM and Bi-LSTM. The output dimensionality of LSTM and Bidirectional LSTM N is the size of $\mathbf{n} \times \mathbf{k}$, where k is the number of distinct semantic role label, and Ni, j corresponds to the score of the j^{th} label of the i^{th} token in the input sequence. For a sequence of predictions y= (y1, y2, y3, … yn), we define the score function as the following:

$$\textbf{Score}\ (\boldsymbol{S}, \boldsymbol{y}) = \Sigma^{n}_{i=0}\boldsymbol{A}_{\mathbf{y},\ \mathbf{y+1}} + \Sigma^{n}_{i=1}\mathbf{N}_{\mathbf{i},\ \mathrm{y}\mathbf{i}}$$

Where A is the transition matrix of CRF such that Ai, j represents the score of the transition from the label i to label j, y0 and yn are the commencing and semantic role label of a sequence. Instead of capturing hand-crafted features, we use context vector output of LSTM and BiLSTM as an input to feature to CRF model that decodes it into the desired name semantic role label. Similarly, the conditional probability of sequence **y** given input **S** turns out to be the following:

$$\boldsymbol{p}(\boldsymbol{y}|\boldsymbol{S}) = \mathbf{e}^{\,\textbf{score}\,(\mathbf{S},\ \mathbf{y})} / \Sigma\boldsymbol{y}\hat{}\in\mathbf{Yx}\ \mathbf{e}^{\,\textbf{score}\,(\mathbf{S},\ \mathbf{y})}$$

Where **Y** is the vocabulary of all possible label sequences of given input **x** throughout the training, we maximize log-likelihood of the correctly labeled sequence as:

$$\boldsymbol{log}(\boldsymbol{p}(\boldsymbol{y}|\boldsymbol{S})) = \boldsymbol{s}\text{core}\ (\boldsymbol{S}, \boldsymbol{y}) - \boldsymbol{log}\ ((\Sigma\boldsymbol{y}\hat{}\in\boldsymbol{Y}\ \boldsymbol{e}^{\boldsymbol{s}\text{core}\ (\boldsymbol{S},\ \boldsymbol{y}\hat{})}\ )$$

$$= \boldsymbol{s}\text{core}\ (\boldsymbol{S}, \boldsymbol{y}) - \underset{\boldsymbol{y}\hat{}\in\boldsymbol{Yx}}{\boldsymbol{logadd}}\ \boldsymbol{s}\text{core}\ (\mathbf{S}, \boldsymbol{y}\hat{})$$

While decoding, we predict an output sequence that achieves the highest scores and predicts the best label path as follows:

$$\boldsymbol{y}* = \underset{\boldsymbol{y}\hat{}\in\boldsymbol{Yx}}{\boldsymbol{argmax}}\ \boldsymbol{s}\textbf{core}\ (\boldsymbol{S}, \boldsymbol{y}\hat{})$$

3.5.3 Training process

Training process is the main part of our architecture. The input for this process is a training file containing words with their feature vector and semantic role label and training process uses different hyper-parameter value (batch size, epoch) as we put in table 7 to train the model, the model building process starts to form a model considering the features and their semantic role label labeled. Features are represented in categorical form is changed to the form of floating numbers by using keras to_categorical method and feed to the embedding layer for embedding every token to fixed size vector as feature representation.

3.5.4 Trained prediction model

The prediction model is the final trained model from the training. The input for prediction is the of words in a given plain Afaan Oromo text or from train test split. By taking this input it predicts the semantic role label of a word, therefore the output is target words with their predicted semantic role as a model evaluation purpose. As we see from this stage it is possible to label and classify sentence constituent with semantic role in Afaan Oromo text using BIO tagging and semantic role we take from PropBank semantic role label.

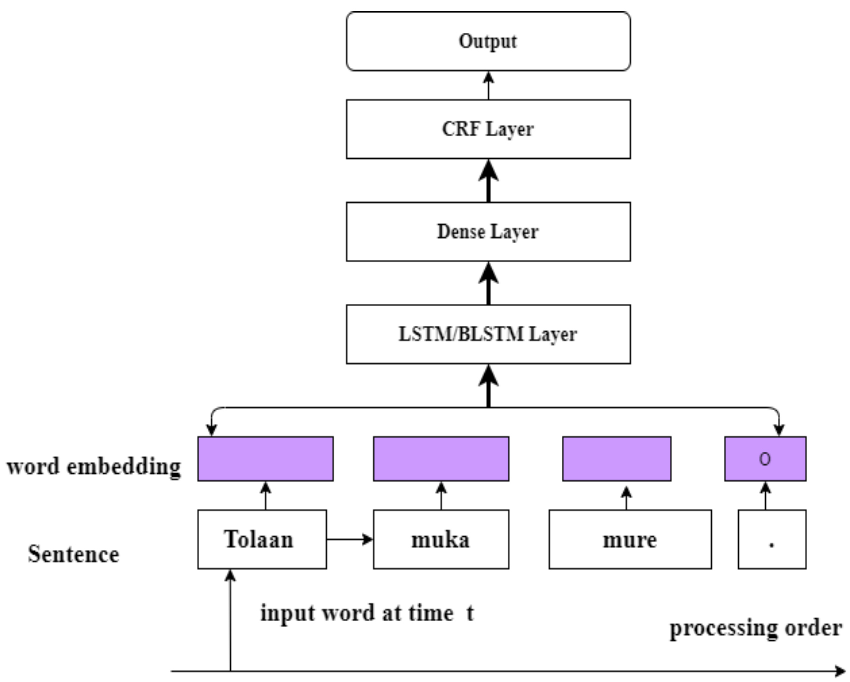


Figure 10: The model architecture with input format

CHAPTER FOUR

4 EXPERIMENTAL RRESULTS AND DISCUSSIONS

In this chapter experimental procedures, datasets, tools and the experimental scenarios used for evaluating our hypothesis are discussed.

4.1 Data Collection

For our study, we have collected 1800 simple Afaan oromo sentences from various sources of Afaan Oromo text manually at random. Those sources are classified as under various domain or categories such as Afaan Oromo previous research corpus. Accordingly, official face book page of (OBN, OMN, FNBC, EBC, BBC Afaan Oromo, Voice of America (Afaan Oromo service)), and websites of (www.fanabc.com Afaan Oromo section is some of the main data sources. In this study the we have used annotated data that area annotated with, semantic role labels by the help of scholars around the domain (i.e., Afaan Oromo language and English). Sentences are labeled manually using SRL label which was developed by [27] based on Afaan Oromo language rule and verified by linguistics of Afaan Oromo language.

4.2 Data Preparation

In our method, however, what we need is just the semantic role labels. Then by the help of linguistic the collected data is annotated with semantic role tag from PropBank and by following the BIO tagging style, then the annotated sentence is converting to vectors by representing each word of the dataset as numerical vectors, and applying padding to adjust all of the sentences to the length of the longest sentence (the sentence with the largest number of tokens). Sentences that are shorter than this length will be filled with zero(s) in the gap. we can make use of these word embedding vectors as an input sequence to a neural network model [28]. Padding allows TensorFlow to ingest data in batches, in which data entries are designed to have an identical fixed length. This procedure makes TensorFlow process data more efficiently. Training of deep learning algorithms needs a large number of datasets, the experiment used 90% of the over-all dataset for training and 10% of the over-all dataset for testing, respectively.

We used PropBank annotation method for this study. Verb is the head node of a sentence. In general, in PropBank propositions verb is correspond to the predicate. The argument structure of a verb is encoded by its complements and modifiers. In PropBank Complements are verbal arguments, that correspond to numbered arguments. Modifiers in parse tree have the same function as modifiers in PropBank propositions and correspond to ARGMs.

Table 4: Basic Relation between Parse tree Node and PropBank Roles

PropBank Label	**Phrase Structure Node**
ARG0 . . . ARGn	Complement
ARGM-xxx	Modifier
Predicate	Head

Example: **Dajaneen laqana nyaate.**

- ✓ **Pos**: -PN Dajaneen NN laaqana
- ✓ **Parsing**: - (NP (PN Dajaneen) (VP (NP (NN laqana)) (VB nyaate)))
- ✓ **Strg:- (‘S(-** (NP (PN Dajaneen) (VP (NP (NN laqana)) (VB nyaate)))’).
- ✓ **Role**:-((**ARG0-AGT**(NP(PN Dajaneen))(VP(**ARG1-PAT**(NN laqana)) (**VB**(TCV nyate)))

After this apply BIO tagging mechanism to tag the sentence with semantic role tag of each word in the sentence with their semantic role.

4.2.1 BIO tagging

We use the BIO tagging style for preparing data for training. The BIO / IOB format (short for inside, outside, beginning) is a standard tagging format for tagging tokens in a chunking task in computational linguistics that we use to prepare data for training (ex. named-entity recognition & semantic role labeling). The **B-** prefix before a tag indicates that the tag is the beginning of a chunk, and an **I-** prefix before a tag indicates that the tag is inside a chunk. The **B-** tag is used only when a tag is followed by a tag of the same type without O tokens between them. An **O** tag indicates that a token belongs to no semantic role/chunk.

Eg1.**Tolaan laaqana nyaate.**

The above simple Afaan Oromo sentence tagged using BIO tagging style as follows

Tolan B-ARG0-AGENT------ doer of the action (who)

Laaqana B-ARG1-PAT-----------receiver of the action (what)

Nyaate B-VB-----it is the predicate(verb)

Eg2. Etiyoo telekom tajajila 4GLT jalqbsise.

Etiyoo B-ARG0-AGT------ doer of the action (who)

Telekom I-ARG0-AGT------ doer of the action (who)

Tajajila O ----outside the chunk

4GLTE O----outside the chunk

Jalqabsise B-VB -----it is the predicate(verb)

Table 5: Numbers of each label in development data.

N0	Class	Number of Instance
1.	ARG0-AGT	1838
2.	ARG1-TEM	1472
3.	ARG1-PAT	1267
4.	ARG1-EXP	22
5.	ARG1-EXT	244
6.	ARG2-BEN	240
7.	ARG2-INS	136
8.	ARG3-SRC	192
9.	ARG4-DES	317
10.	ARGM-PUR	152
11.	ARGM-CAU	262
12.	ARGM-LOC	564
13.	ARGM-TMP	546
14.	ARGM-MNR	175
15.	ARGM-NEG	54
Total = 7481 instances		

4.3 Development tools

Many development tools are used in the process of this study. The tools that have been used include Deep learning library, TensorFlow deep learning library, Keras deep learning library, Scikit learn library and python programming language Python 3.8 programming language installed with anaconda is used to experiment Afaan Oromo AOSRL. We chose to work with Python because of the rich community and library infrastructure. We have listed the Development tools here below in table 6.

Table 6: list of development tools we used

Development tools	**Description**
Anaconda	We used the conda command, we can install Python library and many other packages easily.
Tensor flow	TensorFlow is used as a back end for Keras library which is used for development of deep neural network classifiers
Keras	For the proposed AOSRL system, Keras is used to create deep neural network classifiers.
Scikit learn	Scikit learn is used to evaluate deep neural classifiers by calculating performance metrics

4.4 Experimental setup

One laptop computer is used for the experiments. The machine is used for experimenting deep neural networks using window environment. Hardware and software specification as follows

Table 7: Specifications of machine used for deep neural network experiments

Manufacturer	Hp
Model	OPTIPLEX 7010
Processor	Intel® Core (TM) i3-240 CPU @ 2.9 GHz
Memory (RAM)	4.00 GB (3.89GB usable)
Operating System	Windows 10

4.4.1 Hyper-parameter settings

Hyper-parameter is configured for both LSTM and BiLSTM to train and build model. We implement with different values of batch size, number of units in each layer, optimizers, dropout rate and activation functions. We train our models using the training data and optimizing hyper-parameters on test split. To train our parameter, we mainly based on the back-propagation algorithm which updates parameters on every training example, throughout the time. For parameter optimization and the loss function minimization, we employ relu activation function and Adam optimizer algorithm with recurrent dropout rate of 0.1 for each input to regularization during input. To generalize our model and avoid overfitting, we add a dropout layer with dropout probability of 0.25 just after the embedding layer. The hyper-parameter used to train our model is summarized in Table below.

Table 8: Hyper-Parameter Setting for the Proposed Model (LSTM and BLSTM)

Hyper-parameter description	value
Word embedding dimension	100
Number of iterations(epoch)	20
LSTM units	100 with 2 layers
BLSTM units	100 with 2 layers
Dropout rate	0.25
Optimization algorithm	Adam
Activation algorithm	relu
Batch size	32
vocabulary size	4945

4.5 Experiment using LSTM and BLSTM

We used certain parameters with fixed sizes, such as vocabulary size, embedding Dimension 1activation function, batch size, optimizer, loss=crfloss, dropout and epochs as stated in table 7 above. After testing with different parameter settings of the LSTM and BLSTM models, we recorded the test accuracy, average precision, average recall, and average F1-score, as depict below

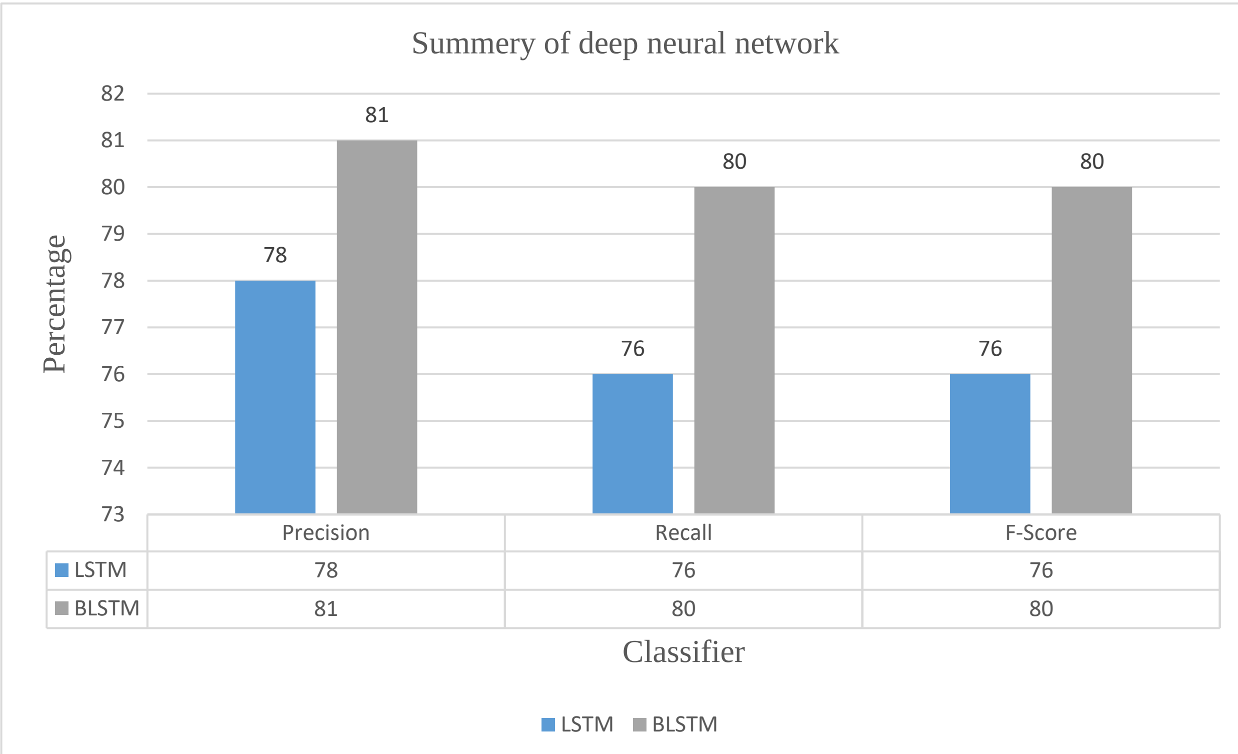


Figure 11: Results from deep neural networks

	precision	recall	f1-score	support
B-ARG0-AGT	0.72	0.85	0.78	116
B-ARG1-EXP	0.00	0.00	0.00	3
B-ARG1-PAT	0.39	0.41	0.40	63
B-ARG1-TEM	0.19	0.18	0.18	50
B-ARG2-BEN	0.00	0.00	0.00	7
B-ARG2-EXT	0.67	0.13	0.22	15
B-ARG2-INS	0.00	0.00	0.00	5
B-ARG3-SRC	0.25	0.27	0.26	11
B-ARG4-DES	0.70	0.39	0.50	18
B-ARGM-CAU	0.00	0.00	0.00	9
B-ARGM-LOC	0.09	0.06	0.07	32
B-ARGM-MNR	1.00	0.25	0.40	12
B-ARGM-NEG	0.00	0.00	0.00	2
B-ARGM-PUR	0.00	0.00	0.00	7
B-ARGM-TMP	0.18	0.30	0.23	33
I-ARG0-AGT	0.38	0.32	0.35	63
I-ARG1-EXP	0.00	0.00	0.00	1
I-ARG1-PAT	0.18	0.37	0.25	38
I-ARG1-TEM	0.26	0.29	0.27	78
I-ARG2-BEN	0.00	0.00	0.00	2
I-ARG2-EXT	0.50	0.18	0.27	11
I-ARG2-INS	0.00	0.00	0.00	5
I-ARG3-SRC	0.17	0.22	0.19	9
I-ARG4-DES	0.77	0.67	0.71	15
I-ARGM-CAU	0.00	0.00	0.00	25
I-ARGM-LOC	0.13	0.15	0.14	39
I-ARGM-MNR	1.00	0.17	0.29	6
I-ARGM-PUR	0.00	0.00	0.00	16
I-ARGM-TMP	0.12	0.42	0.19	24
O	0.92	0.79	0.85	240
PAD	1.00	1.00	1.00	1496
VB	0.98	0.99	0.99	174
accuracy			0.80	2625
macro avg	0.33	0.26	0.27	2625
weighted avg	0.81	0.80	0.80	2625

	precision	recall	f1-score	support
B-ARG0-AGT	0.72	0.80	0.75	123
B-ARG1-EXP	0.00	0.00	0.00	3
B-ARG1-PAT	0.35	0.29	0.32	65
B-ARG1-TEM	0.39	0.31	0.35	74
B-ARG2-BEN	0.00	0.00	0.00	16
B-ARG2-EXT	0.00	0.00	0.00	13
B-ARG2-INS	0.00	0.00	0.00	8
B-ARG3-SRC	0.00	0.00	0.00	6
B-ARG4-DES	0.93	0.82	0.87	17
B-ARGM-CAU	0.36	0.57	0.44	7
B-ARGM-LOC	0.20	0.07	0.11	28
B-ARGM-MNR	0.00	0.00	0.00	5
B-ARGM-NEG	0.00	0.00	0.00	2
B-ARGM-PUR	0.00	0.00	0.00	2
B-ARGM-TMP	0.31	0.32	0.31	41
I-ARG0-AGT	0.24	0.37	0.29	81
I-ARG1-EXP	0.00	0.00	0.00	2
I-ARG1-PAT	0.09	0.30	0.14	50
I-ARG1-TEM	0.21	0.28	0.24	112
I-ARG2-BEN	0.00	0.00	0.00	14
I-ARG2-EXT	0.00	0.00	0.00	10
I-ARG2-INS	0.00	0.00	0.00	3
I-ARG3-SRC	0.00	0.00	0.00	7
I-ARG4-DES	0.79	0.83	0.81	18
I-ARGM-CAU	0.18	0.67	0.28	9
I-ARGM-LOC	0.14	0.10	0.12	31
I-ARGM-MNR	0.00	0.00	0.00	6
I-ARGM-PUR	0.00	0.00	0.00	6
I-ARGM-TMP	0.23	0.27	0.25	33
O	0.87	0.79	0.83	234
PAD	1.00	1.00	1.00	1421
VB	0.90	0.58	0.70	178
accuracy			0.76	2625
macro avg	0.25	0.26	0.24	2625
weighted avg	0.78	0.76	0.76	2625

Figure 12: Classification report for LSTM& BLSTM

4.6 Evaluation

4.6.1 Evaluation metrics

The proposed system has been evaluated using relative metrics such as precision, recall, accuracy and F-score that help to properly measure the performance of the proposed system about how it solves the identified problem. In this study, we evaluate the performance of the model using test data set. A common approach for evaluating machine learning models is through comparison of predicted outputs of the model with that of labeled data by humans. Depending on the similarity between the two, a standard measure used in most researches called F-score. F-score is combined measure of precision and recall. Precision is number of items correctly labeled as belonging to positive class divided by total number of elements labeled as belonging to positive class. Recall is defined as number of true positives divided by total number of elements that actually belong to positive class.

$$\text{Precision} = TP/FP+TP, \qquad \text{Recall} = TP/FN+TP$$

True positive (TP) measures SRL tags which are produced by the model that match the true labels. False Positives (FP) measures SRL tags returned by the model that are not in the true labels and lastly, False Negatives (FN) are SRL tags in the true labels but they are missed by the model.

$$\text{F-Measure} = (\beta^2 + 1)\, PR / (\beta^2 R + P)$$

β^2 is a measure of weighting between precision and recall. An evenly weighted F-measure can be calculated by setting $\beta=1$. Then the previous equation becomes as follows:

$$\text{F-Measure} = (2 * \text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

```
Sample number 52 of 175 (Test Set)
==============================================
Word           ||True(labeled)   ||Prediction
==============================================
galaanan      : B-ARG0-AGT        B-ARG0-AGT
kosartiidhaaf: B-ARGM-PUR        B-ARG2-INS
gara          : B-ARG4-DES        B-ARG4-DES
amerikaa      : I-ARG4-DES        I-ARG4-DES
imale         : VB                VB
.             : O                 O
```

Figure 13: Sample prediction using test data

From the above table, the precision is 5/6 (because there is 5 correct prediction, VB, B-ARG4-DES, I-RG4-ES, B-ARG0-AGT and O among 6 predictions) and the recall is 5/6 (5 correctly found out of 6). For all experiments in this research F-measure is used as a performance evaluation metrics. The overall precision, recall and F-score of a classifier is calculated by using a weighted average. The weighted average is calculated by summing, the multiplication of score with number of instances for each class and dividing the result to the total number of instances.

Loss: -The loss function is called objective function which is used to calculate the loss value. The loss on the training and testing data are show in figure 14 and 15. we use the conditional random field loss fun function to calculate the loss value.

$$\text{Loss} = -\sum_{k=1}^{k} T_k \log y_k$$

In the above equation $\mathbf{T_k}$ represent the actual target data and $\mathbf{Y_k}$ reparent the predicted probability of the target. This is calculated after the activation function transformed the output to a vector.

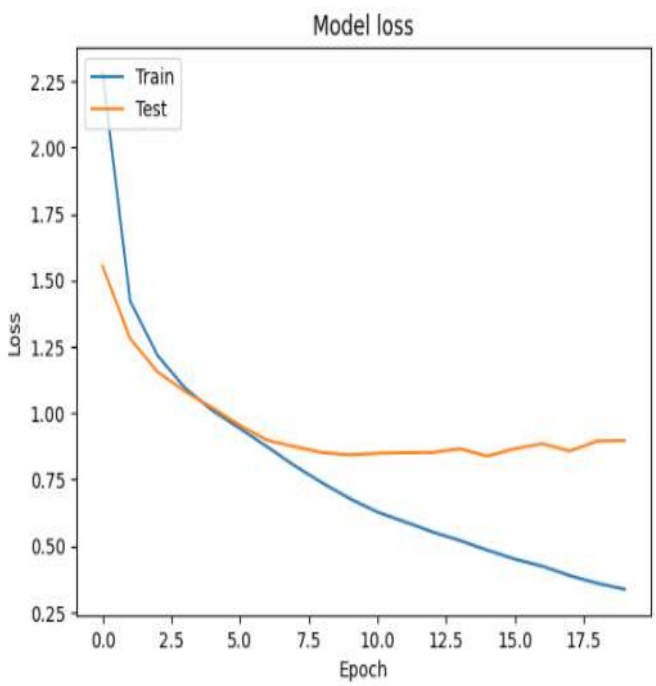


Figure 14: Model loss evaluation for LSTM

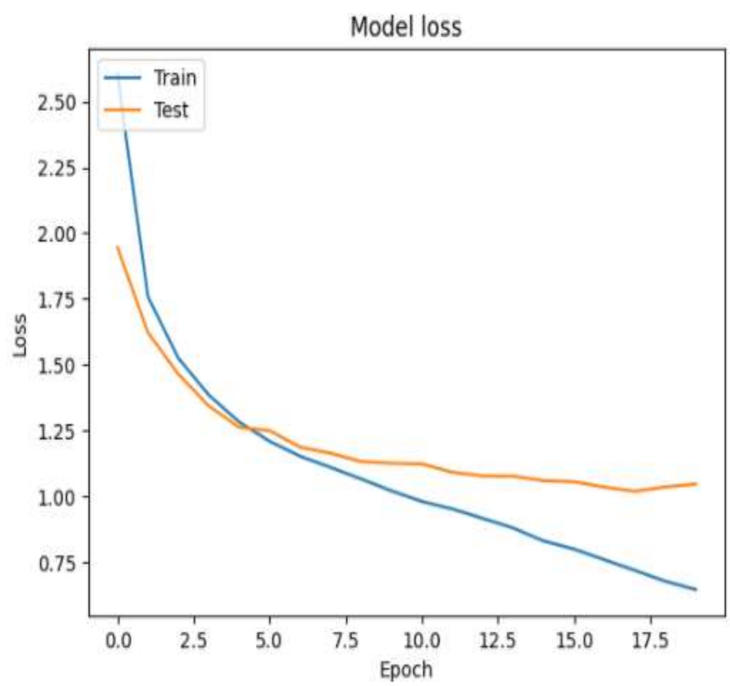


Figure 15 : Model loss evaluation for BLSTM

Figure 14 and 15 shows as the number of epochs is increasing the loss value converges to zero and in the remaining many epochs the line of test data showed above training data. This shows the loss of value of the test data higher than the loss value of the training data.

Accuracy: -The accuracy of the model is calculated by using the Keras accuracy metrics that uses the following formula.

$$\text{Accuracy}=\frac{\text{Amount of the correct predicted instance}}{\text{Total predicted instance}} *100$$

The accuracy of the training and test data showed in figure 16 and 17 below. This show how the semantic role labeling of instance for Afaan Oromo sentence is accurate. If the input is of Afaan Oromo sentence argument and predicate is accurately labeled with semantic role and the model predicted accurate role for new instance within a given training data to the model, since the training of the model is needing a lot of data.

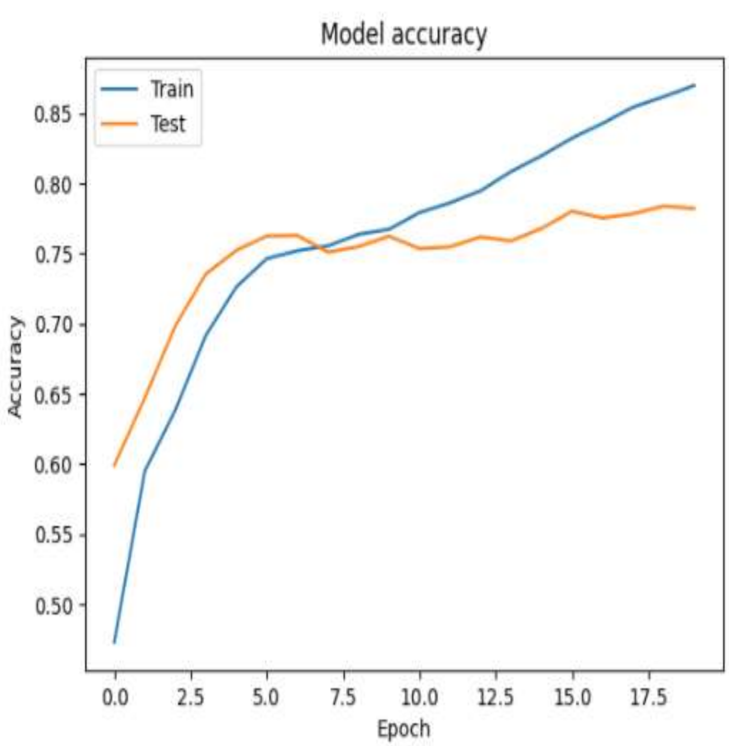


Figure 17: Model accuracy for BLSTM

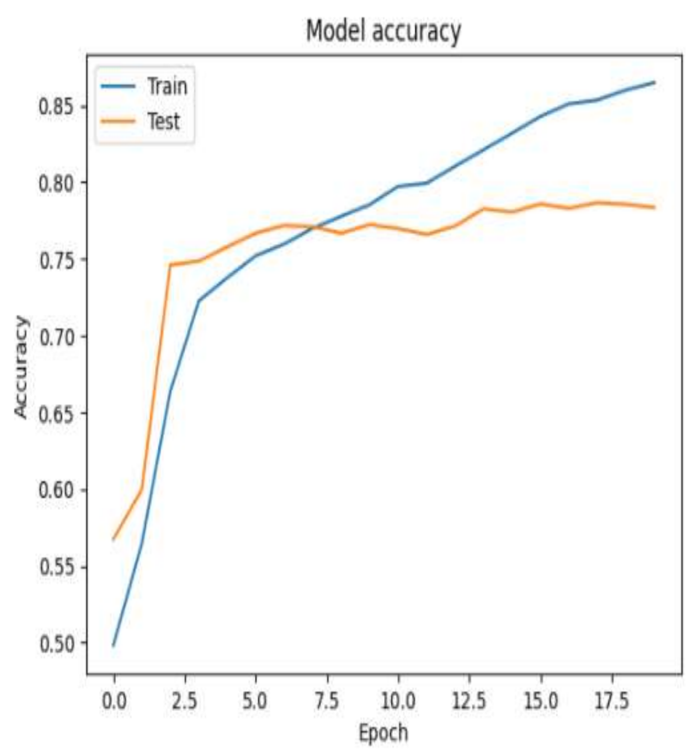


Figure 16: Model accuracy for LSTM

As show in figure 16 and 17 the number of epochs is increasing the accuracy value converges to one. As the first few epochs test data lines are above the training data, this means test data accuracy is greater than accuracy of training and in the remaining many epochs the line of test data showed below the line of the training data. This show the accuracy value of the test data is smaller than the accuracy of the training data. An overall accuracy it was scored is above 70% that is good accuracy, relatively with work of another language, because there is no AOSRL done before for Afaan Oromo, this work is the first work and this is the promising result.

4.7 Discussion

The sample corpus, which was discussed in the above section was used for experimentation. Each sentence in the corpus had been tagged and hand labeled by linguistics and researchers, with comments and suggestions from linguists. However, it was difficult for us to get experts on Afaan Oromo language during the label of semantic role of the corpus. After many times of searching, we are an able to get linguist of Afaan Oromo language those we approach via social media and in person. First. we show to lingues linguistics the way the BIO tagging technique work and semantic role label list that we were used when labeling sentence with semantic role. Then we have sent the corpus that we prepared for this study to them in order to check its correctness.

After that each sentence in our corpus had been tagged and hand labeled correctly by the researchers, based on comments and suggestions from linguists. The sentences in the selected corpus are classified as training dataset and testing dataset. Although, man-made errors were occurred during the training of the system, which are the manual tagging and labeling process were to be one cause for wrongly tagged sentences. So, it was a challenge task that we have been faced. However, we have been evaluated the AOSRL based on semantically labeled Afaan Oromo language depending on the expected output of the labeler and manually tagged/labeled sentences by the researchers based on the linguists' suggestion and help. Besides, there was also challenges during the performance evaluation of the AOSRL, which is because of the absence of a standard criteria to role labeled Afaan Oromo sentences.

We used the result produced by the top-scoring model “Deep Semantic Role Labeling with Self-Attention model “achieves F1 = 83.4 on the CoNLL-2005 shared task dataset and F1 = 82.7 on the CoNLL-2012 shared task dataset, For the development dataset, the F1 score achieved by the by LSTM network was 76, which is 7.4 points lower than the evaluation result from the Self-Attention model system and the F1 score achieved by the bi-directional LSTM network was 80. which is 3.4 points lower than the evaluation result from the Self-Attention model system for the shared task dataset, the best F1 score of our model was 83.4 on the CoNLL-2005 data set while the MATE system achieved 83.4.25. Though the overall performance is not as good as Deep Semantic Role Labeling with Self-Attention, there is still some advantage for the method used in this work. First, the conventional methods such as memory based and rule-based system need careful syntactic feature engineering, which requires much expert knowledge. In contrast, the deep neural network method we used in this work takes the original text as input without any intermediate syntactic representation. It achieves an end-to-end learning process for the SRL task. Furthermore, data we use is small 1800 Afaan Oromo sentence only, while Self-Attention model in millions which means our method can be performed more efficiently if the data is huge properly organized.

Our BLSTM model also produced superior results of f-score 79.77 when compared to model of the, memory-based learning for local Amharic language with a smaller margin, if the Amharic memory-based learning with optimized parameter setting for semantic role labeling task. For the development dataset, the F1 score achieved by the by BLSTM network was 80, which is 0.23 points greater than the evaluation result from the Amharic memory-based learning and the F1 score achieved by the LSTM network was 76. which is 3.77 points lower than the evaluation result from the Amharic memory-based learning model system. Again, our averaged result surpassed their best single model by a margin of 0.23 points. We highlight that the baseline model we used for comparison is reported results only for their best single training sessions with 240 Amharic sentence hand labeled data set. For this reason, we could not produce a direct comparison based on accuracy, data set we use and algorithm we employ, that would possibly produce a more robust evaluation.

CHAPTER FIVE

5 CONCLUSION AND FUTURE WORKS

This chapter is conclusion of observations from our study. It also contains future works to show further study that can be done in the future.

5.1 Conclusion

Semantic role labeling systems are critical application of many NLP applications. Proper semantic role labeling improves the performance of other NLP tasks that rely on it. Different approaches have been used for solving SRL problem which are rule based (linguistic approach), machine learning (statistical approach) and hybrid approach. One researches have been done on SRL for local languages. This re-search used machine learning approach and the features used for classification are manually designed. Designing these features for a given language needs good understanding of language structure. In their training pro-cess combinations of features are tested to come up with best feature set with high performance.

In this work, we investigated Afaan Oromo SRL task using a deep learning method with a particular reference to simple sentences. We have developed four phases of semantic role labeling system for Afaan oromo simple sentence using deep neural network model. The first phase is text pre-processing phase, in this phase the input sentence words are tagged with their semantic role using BIO [27] tagging mechanism. The second stage is model building phase, in this phase we implement three major layer, input, embedding, LSTM/BLSTM, and the dense layer to build the model by taking data from the preprocessing phase, which we manually annotate Afaan Oromo sentence and embed that sentence using Keras embedding layer to train the model and generate word vectors that can capture syntactic and se-mantic relations of words and used it as a feature for our experiments. Then, we construct the classification model & training it, based on the train test split we had set during preprocessing and finally make a prediction with test data to evaluate the model developed.

Despite the fact that deep learning was trained using a large data set, it performed admirably. We only employed a limited data set in our study, but the experiment and model performance evaluation results show that deep semantic role labeler can help with labeling of Afaan Oromo sentence constituents with their semantic roles to certain extent because we get a promising result from both LSTM and BLSTM model.

The system yields promising result, 76% accuracy and 76% F-Score with LSTM and 80% accuracy and 80% F-Score with BLSTM for semantic role belling task. From the two types of LSTM cells, bidirectional LSTM outperforms unidirectional LSTM.Therefore, we can draw a number of conclusions from our investigation of SRL for Afaan Oromo text. First, deep learning can successfully be applied to assign semantic role for Afaan Oromo sentence. Second, properly annotated data are useful for semantic role labeling. As well as attempting optimize deep neural network parameters significantly improve the system performance and from two deep RNN algorithm BLSTM outperform LSTM and this promising result for AOSRL.

5.2 Contributions

The main contributions of this research can be summarized as: -

- ✓ The general architecture of semantic role labeler for Afaan Oromo simple sentence using deep recurrent Neural network (LSTM and BLSTM) algorithm is proposed.
- ✓ We have prepared a semantic role labeled Afaan Oromo sentence that can be used for future researches.

5.3 Future works

This research demonstrated that how to label Afaan Oromo sentence constituent with their semantic role. But to have full edged AOSRL further researches could be conducted. Based on our observations we recommend the following future research areas.

- ✓ When conducting our experiments, we have used small amount of data set to building and training the models. By using a very large data to building and training the model, we believe better results could be achieved.
- ✓ Our study focused on only basic semantic role labeling classes (Core role, adjunct); further researches could consider additional categories of SRL to have complete AOSRL system.
- ✓ We used PropBank, a semantic role labeling method and further study has to be tested the labeling with other methods such as FrameNet and merging approach get comparable results for Afaan Oromo SRL.
- ✓ The scope of this study, simple sentence role labeling is targeted. However, due to existence of different kinds of sentences in Afaan Oromo it is required to extend the labeling for all kinds of sentence as a future work.

References

[1] F. Pascale, Z. Wu, Y. Yang and D. Wu, "Automatic Learning of Chinese English Semantic Structure Mapping," in *In Spoken Language Technology Workshop*, IEEE, 2006, pp. 203-233.

[2] A. Dan and J. Singh,, *A Survey on Semantic Role Labeling and Dependency Parsing,* 2013.

[3] D. Shen, and M. Lapata, ""Using Semantic Roles to Improve Question Answering" In EMNLP-CoNLL," 2007.

[4] G. Stevens, "Automatic Semantic Role Labeling in a Dutch Corpus ,Unpublished Master Thesis,," University Utrecht,, 2007.

[5] I. Titov and A. Klementiev, "Semi-Supervised Semantic Role Labeling: Approaching from an Unsupervised Perspective," in *In Proceedings of the COLING Conference,* 2012.

[6] D. Wu and P. Fung., "Semantic Roles for SMT: A Hybrid Two-Pass Model. In Proceedings of Human Language Technologies," in *The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2009.

[7] D. Liu and D. Gildea, "Semantic Role Features for Machine Translation," in *In Proceedings of the 23rd International Conference on Computational Linguistics, Association for Computational Linguistics*, 2010.

[8] L. Marquez, X. Carreras, K. C. Litkowski and S. Stevenson, "Semantic Role Labeling: An Introduction to the Special Issue”, Computational Linguistics," 2008.

[9] P. Exner and Pierre Nugues, "Using Semantic Role Labeling to Extract Events from Wikipedia," in *In Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011),*, 2014.

[10] D. Gildea and D. Jurafsky, "Automatic Labeling of Semantic Roles," *Computational Linguistics,* vol. 28, no. 3, pp. 245-288, 2002.

[11] H. Sun and D. Jurafsky, "Shallow Semantic Parsing of Chinese," in *In Proceedings of NAACL-HLT,* 2004.

[12] M. Teshome, "Design of Anaphora Resolution for Afaan Oromo personal Pronoun," *unpublished,* 2018.

[13] Girma Debela dinagge & Martha Yifru Tachbelie, "Afaan Oromo news Text Summarizer," *international jornal of computer application,* vol. 103, no. 3, 2014.

[14] C. F. Elenso, "Afaan Oromo list, definition and quastion answering system," *thesis bank,* 2016.

[15] Million Meshesha,yiyayew Solomon, "English Afaan Oromo stastical Machine Translation," *International Journal of Computational linguistic* , 2018.

[16] D. Jurafsky and J. H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition," Boulder,, University of Colorado, 2015.

[17] Y. B. TESEMA, "Hybrid Word Sense Disambiguation Approach for Afaan Oromo," 2016.

[18] D. Megersa, "An Automatic Sentence Parser for Oromo Language Using top down approach," Addis Ababa University, Addis Ababa, 2002.

[19] D. J. H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition," *University of Colorado,* 2015.

[20] Rui Pedro Talhadas dos Santos, *Automatic semantic role labeling for European Portuguese,* 2014.

[21] D. Shen, and M. Lapata, "Using Semantic Roles to Improve Question Answering," in *In EMNLP-CoNLL,* 2007.

[22] D Jurafsky H Martin, "Speech and Language Processing: An Introduction to Natural Language Processing," in *University of Colorado,* Boulder, 2015.

[23] A. Taani, M Msallam, S Wedian,, "A Top-Down Chart Parser for Analyzing Arabic Sentences," *The International Arab Journal of Information Technology,* vol. 9, no. 2, 2015.

[24] Charles J. Fillmore and baker, "framenet devlopment for semantic libeling," 1976,1977,1982,1985.2001,2010.

[25] G. Stevens, "Automatic Semantic Role Labeling in a Dutch Corpus," Unpublished Master Thesis, University Utrecht, 2007.

[26] Sue Atkins, Rundell, Michael, and Sato, Hiroaki, "fnbibliography," The framenet constructicon., 10 2012. [Online]. Available: https://framenet.icsi.berkeley.edu/fndrupal/WhatIsFrameNet. [Accessed 02 10 2020].

[27] M. Palmer, D. Gildea and P. Kingsbury, "the Proposition Bank: An Annotated Corpus of Semantic Roles," *Computational Linguistics*, vol. 31, no. 1, 2005.

[28] S. Pradhan, W. Ward, K. Hacioglu, J. Martin and D. Jurafsky,, "”Shallow Semantic Parsing using Support Vector Machines”," *In Proceedings of HLT/NAACL,* 2004.

[29] Wermter Ellen Rilo Gabriele Scheler Springer, "Connectionist, Statistical and Symbolic Approaches to Learning for Natural Language Processing," *Springer Science & Business Media,* vol. 1040, 1996.

[30] K. Shaalan, "Rule-based Approach in Arabic Natural Language Processing," *The International Journal on Information and Communication Technologies (IJICT),*, vol. 3, no. 3, 2010.

[31] Cd Manning and H Schütze, "Foundations of Statistical Natural Language Processing," *ambridge: MIT press,* vol. 99, 1999.

[32] A. Dan and J. Singh, "A Survey on Semantic Role Labeling and Dependency Parsing," 2013.

[33] R. Swier and S. Stevenson, "Unsupervised Semantic Role Labeling," in *In Proceedings of EMNLP*, 2004.

[34] V Hagen Fürstenau and B Vorgelegt, "Semi-supervised Semantic Role Labeling via Graph Alignment," in *University at des Saarlandes*, 2011.

[35] H. a. Bengio, "Deep learning," *Nature*, vol. 512, no. 7553, pp. 436 -444, 2015.

[36] J. M. Zurada, " Introduction to artificial neural systems," *West St.*, vol. 8, 1992.

[37] S. H. a. J. Schmidhuber, "Long short-term memory," *Neural compution* , vol. 9, no. 8, p. 1735–1780, 1997.

[38] X. Wang, "Semantic Role Labeling in Chinese Using HowNet," *Language and Linguistics*, vol. 9, no. 2, 2008.

[39] A. K. Ghalibaf, S. Rahati and A. Estaji, "Shallow Semantic Parsing of Persian Sentences," in *PACLIC*, 2009.

[40] E. Yirga, "Semantic Role Labeler for Amharic Text Using Memory Based Learning," 2017.

[41] K. Wang, "Automatic Semantic Role Labeling for Chinese," in *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, 2010.

[42] Daniel Henrique Mourao , Marco Antonio Calijorne Soares ,Wladmir Cardoso Brand and Fernando Silva Parreiras, "Using Recurrent Neural Networks for Semantic Role Labeling in Portuguese," *Springer*, pp. 2-14, October 2019.

[43] Emma Strubell, Patrick Verga, Daniel Andor, David Weiss and Andrew McCallum, "Linguistically-Informed Self-Attention for Semantic Role Labeling," 4 2018.

[44] Jeffrey Pennington, Richard Socher, and Christopher, "Global vectors," In EMNLP, 2014.

[45] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt, "Deep contextualized word representations," In NAACl, 2018.

[46] Timothy Dozat and Christopher D. Manning, "Deep biaffine attention for neural dependency parsing.," In ICLR, 2017.

[47] Kingma, D., Ba, J., "Adam: a method for stochastic optimization," rXiv preprint, 2014.

[48] Srivastava, N., Hinton, Krizhevsky,Sutskever, "Dropout: a simple way to prevent neural networks from overfitting," (2014).

[49] S. C. a. D. Mikolov, "Distributed representations of words and phrases and their compositionality," *in Advances in neural information processing systems,* p. 3111–3119, 2013.

[50] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp. 1532–1543, 2014.*, 2014..

[51] Mikolov, Chen, Corrado, and J. Dean, "“Efficient estimation of word representations in vector space," 2013.

[52] J. Lafferty, A. McCallum, and F. Pereira, “, "Conditional random fields: Probabilistic Probabilistic for segmenting and labeling sequence data," 2001.

[53] Shotaro Misawa, Motoki Taniguchi, Yasuhide Miura and Tomoko Ohkuma, "Character-based Bidirectional LSTM-CRF with words and characters for Japanese Named Entity Recognition," in *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, Copenhagen, Denmark, 2017.

[54] "Application of a Hybrid Bi-LSTM-CRF model to the task of Russian Named Entity Recognition," *Communications in Computer and Information Science,* vol. 78, 2018.

[55] G. S. P. C. Pierpaolo Basile, "Bi-directional LSTM-CNNs-CRF for Italian Sequence Labeling," in *Core Clic-It 2017*, 2017.

[56] Fox, "keras deep learning libray," 2016.

[57] S. Pradhan, W. Ward, K. Hacioglu, J. Martin and D. Jurafsky, "Shallow Semantic Parsing using Support Vector Machines," in *In Proceedings of HLT/NAAC,* 2004.

[58] K. Wang, "Automatic Semantic Role Labeling for Chinese," 201.

[59] M. Aronoff and K. Fudeman, "What is Morphology," *Blackwell Publishing,* vol. 8, no. 1, 2011.

[60] S Pradhan W Ward K Hacioglu J Martin and D Jurafsky, "Shallow Semantic Parsing using Support Vector Machines," in *In Proceedings of HLT/NAACL,*, 2004.

[61] Xue, N. and Palmer, M, "Calibrating features for semantic role labeling," in *In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP),*, 2004.

[62] Matthes, E., Python Crash Course: A Hands-on, Project-Based Introduction to Programming., San Francisco: No Starch Press, 1972.

[63] M. E, "A Hands-on, Project-Based Introduction to Programming.," 1972.

[64] N. Srivastava, Hinton. Krizhevsky. Sutskever and Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of machine learning research,* vol. 15, no. 1, p. 1929–1958, 2014.

[65] H. Beshada, "Design and devlopment of senetence parser for Afaan Oromo," *Design and devolope senence parser for Afaan Oromo language using top down approach,* p. 36, 2017.

[66] AYANA, ABRAHAM GIZAW, "Improving Brill’s Tagger Lexical And Transformation Rule For Afaan Oromo Language," *addis abab university,* 2017.

Appendix A: Sample Data preparation by using BIO tagging

108	yunivarsiitin	B-ARG0-AGT
108	finfinne	I-ARGO-AGT
108	artist	B-ARG2-BEN
108	alii	I-ARG2-BEN
108	birrattif	I-ARG2-BEN
108	docteereta	B-ARG1-TEM
108	kabajaa	I-ARG1-TEM
108	kennef	B-VB
108	.	O
109	gondaritti	B-ARGM-LOC
109	pirojectiin	B-ARG1-TEM
109	bishaanii	I-ARG1-TEM
109	har'a	B-ARGM-TMP
109	ebbifame	B-VB
109	.	O
110	tolan	B-ARG0-AGT
110	hatattamaan	B-ARGM-MNR
110	gara	B-ARG4-DES
110	manaa	I-ARG4-DES
110	debihe	B-VB
110	.	O
111	sababa	B-ARGM-CAU
111	hanqina	I-ARGM-CAU
111	chekitin	I-ARGM-CAU
111	mindaan	B-ARG1-TEM
111	hinkaffalmne	B-VB
111	.	O
112	sababa	B-ARGM-CAU
112	jequmsaatin	B-ARGM-CAU
112	karaan	B-ARG1-PAT
112	cufame	B-VB
112	.	O
113	qonnan	B-ARG0-AGT
113	bulaan	I-ARGO-AGT
113	omisha	B-ARG1-TEM
113	isaa	I-ARG1-TEM
113	maasirra	B-ARGM-LOC
113	gara	B-ARG2-DES
113	gabayaa	I-ARG2-DES
113	gesse	B-VB

114	barnoonni	B-ARG1-TEM
114	bara	B-ARG1-TEM
114	2O12	I-ARG1-TEM
114	bor	B-ARGM-TMP
114	jalqaba	B-VB
114	.	O
115	abiya	B-ARG0-AGT
115	ahimad	B-ARG0-AGT
115	badhaasa	B-ARG1-TEM
115	nobeelii	I-ARG1-TEM
115	argate	B-VB
115	.	O
116	vaayirasii	B-ARG1-TEM
116	koronaa	I-ARG1-TEM
116	hirdhisuuf	B-ARGM-PUR
116	sochiin	B-ARGM-MNR
116	jalqabame	B-VB
116	.	O
117	kaasan	B-ARG0-AGT
117	laaqana	B-ARG1-PAT
117	isaa	I-ARG1-PAT
117	nyaate	B-VB
117	.	O
118	dabalaan	B-ARG0-AGT
118	asiin	B-ARGM-LOC
118	darbe	B-VB
118	.	O
119	jaldessi	B-ARG0-AGT
119	boqqollo	B-ARG1-PAT
119	ballese	B-VB
119	.	O
120	manni	B-ARG1-TEM
120	keenya	I-ARG1-TEM
120	karaa	B-ARGM-LOC
120	duraadha	B-VB
120	.	O

Appendix B: List of PropBank Semantic Role Used for Data Annotation

N0	Semantic Role	Description
16.	ARG0-AGT	Agent (doer of the action)/who
17.	ARG1-TEM	Theme (receiver of the action but state doesn’t change)/who
18.	ARG1-PAT	Patient (receiver of the action and state changed) what
19.	ARG1/M-EXP	Experiencer
20.	ARG2-EXT	Extension(size/percent)
21.	ARG2-BEN	Beneficiary (beneficiary of the action)
22.	ARG2-INS	Instrument (by what)
23.	ARG3-SRC	Source (starting point)
24.	ARG4-DES	Direction or Goal / Destination (end point)
25.	ARGM-PUR	Purpose (for what)
26.	ARGM-CAU	Cause (because)
27.	ARGM-LOC	Location (where)
28.	ARGM-TMP	Time (when)
29.	ARGM-MNR	Manner (how)
30.	ARGM-COM	Comitative (with what)
31.	ARGM-NEG	Negation (not)

Appendix C: Summery of the model for LSTM

```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 15)                0
_________________________________________________________________
embedding_1 (Embedding)      (None, 15, 100)           494700
_________________________________________________________________
lstm_1 (LSTM)                (None, 15, 100)           80400
_________________________________________________________________
lstm_2 (LSTM)                (None, 15, 100)           80400
_________________________________________________________________
dropout_1 (Dropout)          (None, 15, 100)           0
_________________________________________________________________
time_distributed_1 (TimeDist (None, 15, 33)            3333
_________________________________________________________________
crf_1 (CRF)                  (None, 15, 33)            2277
=================================================================
Total params: 661,110
Trainable params: 661,110
Non-trainable params: 0
_________________________________________________________________
```

Appendix D: Summery of the model for BLSTM

```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 15)                0
_________________________________________________________________
embedding_1 (Embedding)      (None, 15, 100)           494700
_________________________________________________________________
bidirectional_1 (Bidirection (None, 15, 200)           160800
_________________________________________________________________
bidirectional_2 (Bidirection (None, 15, 200)           240800
_________________________________________________________________
dropout_1 (Dropout)          (None, 15, 200)           0
_________________________________________________________________
time_distributed_1 (TimeDist (None, 15, 33)            6633
_________________________________________________________________
crf_1 (CRF)                  (None, 15, 33)            2277
=================================================================
Total params: 905,210
Trainable params: 905,210
Non-trainable params: 0
_________________________________________________________________
```