Thesis/Project No: CSER-20-27

A Shallow Parser for Bangla

By

Md. Rubel Hasan

Roll: 1507074

&

Mikail Biswas Mridu

Roll: 1507081

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

February, 2020

A Shallow Parser for Bangla

By

Md. Rubel Hasan

Roll: 1507074

&

Mikail Biswas Mridu

Roll: 1507081

Project/Thesis, Course No: CSE4000

Supervisor:

Mohammad Insanur Rahman Shuvo

Assistant Professor

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

_____

Signature

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

February, 2020

Acknowledgement

At first, we would like to thank Almighty God for showering all his blessings on us whenever we needed. It is our great pleasure to express our indebtedness and deep sense of gratitude to our Supervisor Mohammad Insanur Rahman Shuvo, Assistant Professor, Department of Computer Science and Engineering (CSE), Khulna University of Engineering & Technology (KUET) for his continuous encouragement, constant guidance, great and enthusiastic support, both during the implementation phase and the writing process and keen supervision throughout the course of this study. His prudence and ethics have been a great inspiration for the work. We also like to remember the inspiration, supports and encouragement of my loving family.

We are extremely grateful to all the faculty members of the Department of CSE, KUET to have their privilege of intensive, in-depth interaction and suggestions for the successful completion of our thesis work.

February, 2020 Authors.

Abstract

In natural language processing or text mining, partial parsing or shallow parsing holds a very prominent role of extracting syntactic information from natural languages that can help in solving many more natural language processing problems. This thesis work is basically an effort to build a shallow parser for Bengali language, which consists of two major parts – a maximum entropy model-based parts-of-speech (POS) tagging and a rule based chunker. In this thesis work, we used 16 POS tags to label the constituent words in Bengali sentence and some hand written rules to chunk the sentence. The POS tagger categorizes each token in Bengali test sentence and a rule based chunker chunks the sentence in word groups that are syntactically correlated. Using of machine learning based approach like maximum entropy model for POS tagging significantly increases the accuracy of POS tagging. For information extraction, information retrieval or machine translation of Bengali language, the shallow parser can be of great help.

Contents

List of Figures

Vii

List of Tables

vii

Chapter 1

Introduction

1.1 Objective

Natural languages are the most common way for the communication between human beings. In this age of digital communication, the revolution of communication system increasing through the use of natural language text over internet-based systems. Over the internet, huge amount of natural language-based data is available and continuously growing as the time passes on. This imposes the necessity of finding an easy way to process the extreme amount of textual data to extract relevant information while understanding the context of these data. All these data are natural language based, so natural language processing is a must. To understand and process natural language sentence, extensive contextual knowledge about natural languages and manipulation of it is highly needed. Teaching computer about the nature of natural language processing is extremely difficult. Computers cannot easily understand the rules that dictate information transmissions via natural languages. Algorithms capable of parsing simple natural language text sentences and determining syntax information exists, but they mostly fail to process complex natural language text sentences as no complete grammar for any natural languages exists. This necessitates the process of computational aspects of natural languages to extract relevant information. Shallow parsing can be proved highly practical in this aspect of natural language processing.

1.2 Overview of Shallow Parsing

Parsing is basically transforming the sentences into a representation of groups of words or phrases and their relations among them. To parse a Bengali sentence, we will need a context-free-grammar (CFG) to completely finish the parsing process. But sometimes many NLP applications such as information retrieval, information extraction or machine translation require only partial syntactic information. So, we can generate a partial structure of sentence that can easily extract the syntactic information that we need. The complete parse tree generation for complex sentences is expensive for linguistic computations whereas partial parse tree always in-corporate tokens with depth two, which is exceptionally economical for linguistic computations.

The example of a complete parse tree using CFG of Bengali language is illustrated bellow:

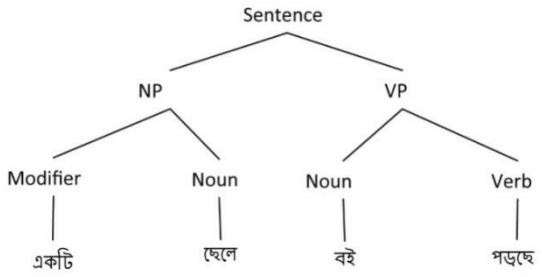

Figure 1.1: Complete parsing of a simple sentence using Bengali CGF [8]

Shallow parsing is basically one kind of syntactical analysis of a sentence that determines the constituent pats of the sentence such as noun, verb, adjective etc. and linking them to some discrete grammatical groups like noun phrases or verb phrases and then chunking them using some regular expressions and all these process can better reflect the semantic relations with the use of machine learning models for extracting contextual information of the tokens in sentences[6]. Shallow

parsing gives the partial structure of the parse tree of sentence, for example: the shallow parsing of a simple Bengali sentence will be like –

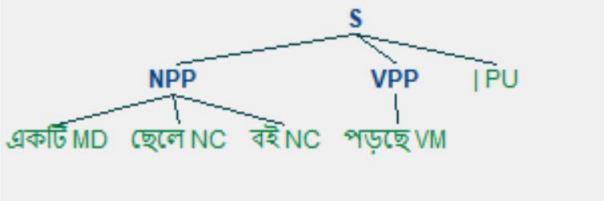

Figure 1.2: Shallow parsing of a simple sentence

The first step of shallow parsing of a Bengali sentence - POS tagging isn't a very easy job to be done because the POS tag for any word not only depends on the word but also on it's context in the sentence as the same word can be a noun(বিশেষ্য) or maybe a verb(ক্রিয়া) according to the context of particular sentence. That's why POS tagging of sentences is practical matter of processing through ambiguities.

The second step of shallow parsing of a Bengali sentence – chunking Bengali sentence into some particular grammatical groups that can be used in different NLP applications. The behavior of natural languages is ambiguous, so machine learning approaches can extract useful features efficiently to complete the natural language sentence processing.

1.3 Scope of this thesis work

In our thesis we have worked for Bangla language sentences to label them appropriate parts-of-speech tags and then chunk the tagged sentence into discrete grammatical groups. As natural language processing is a field of computer science, artificial intelligence and the interaction between human and computer through natural language which is based on syntactical and semantical information of particular natural language. Our thesis work has verity of application –

- Named entity recognition
- Relation extraction
- Information retrieval
- Machine translation etc.

1.4 Organization of this thesis work

The remaining part of this thesis consists of the parts as follows:

Chapter 2: Representations of some literature related to POS tagging and chunking of Bengali sentences are reviewed in this chapter.

Chapter 3: The corpus data manipulation, POS tagging and chunking of test input Bengali sentence is explained in this chapter.

Chapter 4: The result and evaluation according some input examples are explained in this chapter.

Chapter 5: The future scope of this thesis work and the conclusive words about the method of POS tagging and chunking are outlined in this chapter.

Chapter 2

Literature Review

2.1 Background

Bengali (or Bangla) is an Indo-Aryan language which is one of the main primarily speaking language of Bangladesh. It is the most commonly spoken and official language of Bangladesh and second most commonly speaking language of India. There are nearly 260 million Bengali speakers where 230 million people are native speakers. That makes Bengali language, the seventh most widely speaking language of the world and fifth most commonly speaking native spoken language. Bengali language processing is at early stage of developing stage of it. For being such a vast language, completely parsing of Bengali language sentences is a bit of costly for linguistics computations. Shallow parsing of Bengali language can be an effective component for different kinds NLP applications of Bengali language. Developing of a Bengali shallow parser or partial parser will impose great influence on different pipelined modules of Bengali language analyzing system including information extraction and retrieval, machine translation. Shallow parsing process of a text sentence is the division of non-recursive syntactical units such as noun phrases, verb phrases, etc. Non recursive implying that phrases or chunks are non-overlapping and do not contain each other as proposed by Abney (1991) [2].

2.2 Related works

There are some literature works that are carried out for maximum entropy-based POS tagging and chunking Bengali language sentences –

- In 2007, Sandipan Dandapat developed a Bengali POS tagger with 40 tags using maximum entropy model while solving Bengali language ambiguities in which tagging accuracy yields to 88.08% with morphology as features [5].
- In 2008, a maximum entropy based Bengali POS tagger with 26 tags were developed by Asif Ekbal, Rejwanul Haque, Sivaji Bandyopadhyay, which yields an accuracy of 88.2% [3].
- In 2006, Sivaji Bandyopadhyay and Asif Ekbal developed HMM based POS tagger and chunker for Bengali which demonstrated 85.85% tagging accuracy and 96.9% chunking accuracy and 81.61% chunking accuracy after POS tagging [7].

There are some attempts to build a shallow parser for Tamil in 2014 [1] and a shallow parser for Khashi in 2018 [4]. Comparing to other languages a little amount of research has been attempted over shallow parsing of Bengali language and rarely any attempt has been made to make a shallow parser based on maximum entropy model-based parts-of-speech tagging and chunking tagged Bengali sentence. So are attempting to make s shallow parser for Bangla consisting of maximum entropy model-based parts-of-speech tagging and chunking the tagged Bengali sentence.

Chapter 3

Proposed Methodology

The machine learning approaches in solving natural language processing problems are becoming popular as days goes on, as machine learning models can automatically and effectively learn from corpus that are annotated correctly. As the required formatted data for shallow parsing is hard to find so we manually made few sentences that can be used as train data for maximum entropy model and correctly classified to parts-of speech tags and we defined some syntactical rules to chunk the tagged data.

3.1 Implementation

We developed the partial parser using Python based IDE PyCharm where we manually defined the maximum entropy trainer for Bengali language using NLTK toolkits and the hand written chunk rules were parsed using Regular expression parser from NLTK. The complete methodology for the Shallow Parsing process is illustrated bellow:

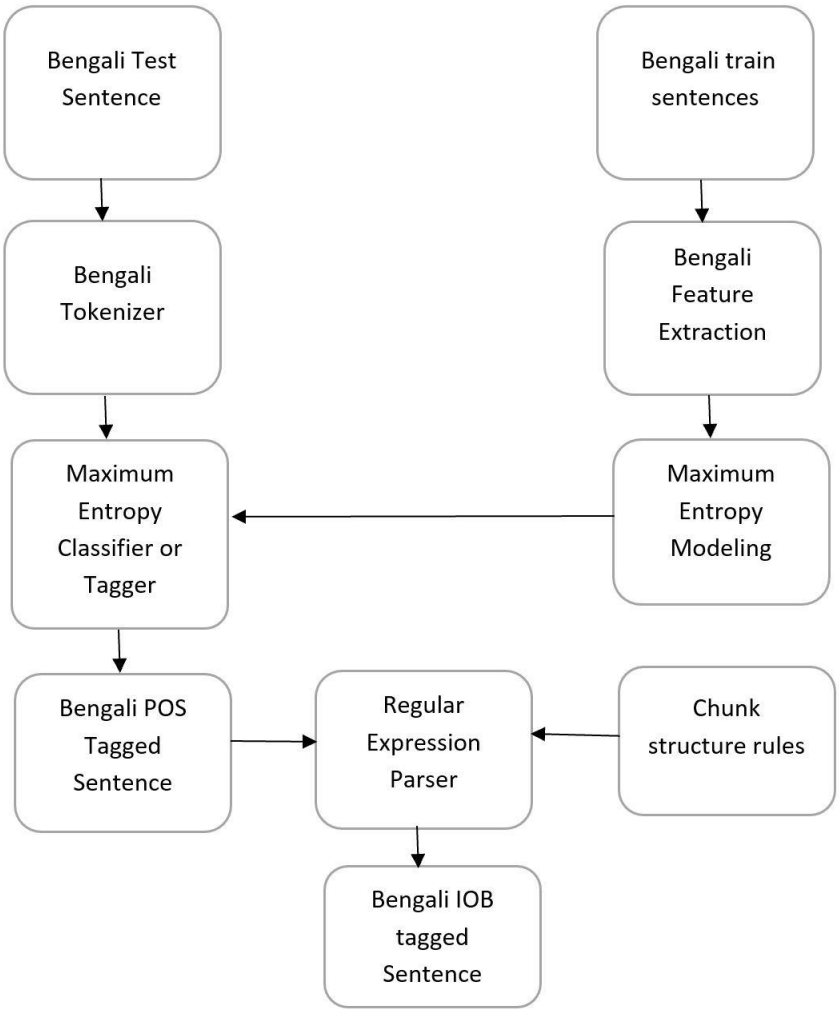

Figure 3.1: Shallow Parsing Process of Bengali Sentence

Basically, at first, we tokenize test input and then send it to maximum entropy-based POS tagger that classify the tokens of input according to their extractable features and POS tagged format of the input sentence is generated. Finally, the POS tagged sentence is chunked using NLTK

Regular Expression parser and IOB tagged format of the input sentence and partially parse tree are generated which can be used other NLP applications.

3.2 Corpus data

The corpus data are prepared by manually tagging at each token existing in the sentences where tokens are given tags by separating with a backward-slash (\). The example of tagged sentences in corpus are like –

আমার\PN সোনার\NC বাংলা\NP ,\PU আমি\PN তোমায়\PN ভালোবাসি\VM |\PU

সততা\NC একটি\MD মহৎ\AJ গুণ\NC |\PU

আমি\PN তার\PN জন্য\PP অপেক্ষা\NC করছি\VM |\PU

তার\PN শরীরে\NC ভিটামিন\NC A\RD -\PU এর\PP অভাব\NC আছে\VM |\PU

আহা\IJ !\PU লোকটি\NC দেখতে\VM পায়\VA না\NG |\PU

সে\PN কি\QQ আসবে\VM ?\PU

3.3 Pre-processing

The corpus data are tokenized after reading from corpus to make a sentence as a list of tuples of tokens, Pos tags. The example of tagged sentence and pre-processed form of it is like-

Sentence in corpus -

আমার\PN সোনার\NC বাংলা\NP ,\PU আমি\PN তোমায়\PN ভালোবাসি\VM |\PU

After pre-processing the sentence –

[('আমার', 'PN'), ('সোনার', 'NC'), ('বাংলা', 'NP'), (',', 'PU'), ('আমি', 'PN'), ('তোমায়', 'PN'), ('ভালোবাসি', 'VM'), ('|', 'PU')]

All the sentences in corpus are loaded in this form to be trained in maximum entropy model where every sentence is list of tuples (tokens, POS-tag) and all the sentences are included in another list. The final list is used as train set for maximum entropy model for extracting useful features and then categorize the test input tokens according to the features extracted from train set.

3.4 Proposed POS tags

After analyzing different tag-sets for Bengali language, we considered 16 tags that will grammatically categorize the test input sentence. The parts-of-speech tag-set description is given bellow:

Table 1: Our POS Tag set

Serial No.	Description	POS tag	Example
1	Proper Noun	NP	বাংলা, রাজা
2	Common Noun	NC	সততা, ভাত
3	Pronoun	PN	সে, আমাকে
4	Adjective	AJ	মহৎ, পরিশ্রমী
5	Main Verb	VM	খাচ্ছে, করছি
6	Auxiliary Verb	VA	পায়, পারে
7	Adverb	AV	দ্রুত
8	Connectives	CJ	ও, তাই
9	Interjection	IJ	আহা, ছিঃ
10	Punctuation	PU	| , -
11	Negative words	NG	না
12	Quantifier	QT	পাঁচ, হাজার
13	Modifier	MD	একটি
14	Residuals	RD	Foreign words
15	Preposition	PP	এর, কাছে
16	Question related words	QQ	কি

3.5 POS Tagging

3.5.1 Maximum Entropy Modeling

Maximum entropy modeling is one kind of statistical modeling technique which has become very popular in recent days, which was used for predicting the appropriate tag for the tokens. Maximum entropy model extracts some features based on some syntactical relations according to their probabilities. The machine learning optimizer takes the most probable features into account for final classification. In this type of linguistic classification supervised machine learning processes are proven to provide with good results as they can extract most of the syntactic information from diverse data set and efficiently trains relative amount of data set and predict the linguistic classes.

Maximum entropy classifier is one kind of framework that contemplates all of the probability distributions that are empirically consistent with the training sentences and picks out the distribution with the highest entropy. A probability distribution is empirically consistent with a set of training data if its estimated frequency with which a class and a feature vector value co-occur is equal to the actual frequency in the data.

A maximum entropy classifier or conditional exponential classifier is parameterized by a set of "weights", which are used to combine the joint-features that are generated from a feature set by an "encoding". In particular, the encoding maps each (feature set, tag) pair to a vector. The probability of each label is then computed using the following equation:

$$\text{prob (words| label)} = \frac{\text{dot-product (weights, encode (words, tag))}}{\text{sum (dot-product (weights, encode (words, tag)) for l in tags)}}$$

Where dot-product means dot-product (a, b) = sum (x*y for (x, y) in zip (a, b)) [9].

The maximum entropy training is done based on rare and non-rare words feature extraction where rare word cutoff lessens the computational cost by a little amount.

3.5.2 Megam Optimization Algorithm

Megam is an optimization algorithm based on CG and LM-BFGS Optimization of Logistic Regression. This technique is faster than iterative scaling algorithm or generalized iterative scaling algorithm [10].

3.6 Features for Bangla

Determining the appropriate features for linguistic test data is of great importance for classifying the tokens with correct POS tags. We consider some cases for feature extracting like –

- For non-rare words, previous two words and their tags and next two words for current word in sentence, which helps to solve word sense disambiguation
- For rare and unseen words, suffix information and prefix information of the current word in sentence, which helps to classify rare unseen words
- Tag dictionary generated from the tags appeared in train data, which helps to determine tags that are need to be considered for machine learning optimization
- If unseen word contain numerals or other signs are considered for feature
- For rare word cutoff we used 1 because the corpus is small

3.7 Chunking

Chunking is the final step of shallow parsing, which will give data that can be used for other NLP applications. We are using a rule based chunker that works for the train data, we used for training in maximum entropy model-based POS tagging process. For chunking we used some hand written rules based on the train data. We are using noun phrase chunk and verb phrase chunk for making chunking rules.

Noun phrase chunking rule is like –

NPP: {<NP|NC>*<MD><AJ>*<NP|NC>*}

{<PN>*<NP|NC>*<PN>*}

Verb phrase chunking rule is like –

VPP: {<VM>*<VA>?}

For chunking we used the following process:

- First, we set syntactic structure noun phrase and verb phrase using NLTK Regular expression parser
- Then NLTK Regular expression parser process the POS tagged data and gives chunked data
- Then chunker is evaluated using chunked format of train data

After chunking POS tagged data, the chunked data is represented by IOB notation where each chunk type is encoded by B – Begin, I – Inside notations and other parts of sentence that are not included in any chunks are encoded by O – Outside notation.

Chapter 4

Experimental Results and Evaluation

Now every time when a new input test data is given to our shallow parser, we can tokenize or pre-process the test sentence, then the tokenized sentence will be given POS tag using maximum entropy based POS tagger and then the Pos tagged sentence will be chunked into syntactical chunk groups like noun phrase or verb phrase and finally the chunked data will be represented by IOB notations. For our POS tagger we have achieved nearly 90.78% accuracy and after chunking POS tagged data, we achieved nearly 88% accuracy, the chunking accuracy was computed by comparing manual generated output and chunker outputs.

The example of some test data processing through our shallow parser is illustrated bellow:

Input sentence-1: আমি তোমায় ভালোবাসি |

POS tagged sentence-1: [('আমি', 'PN'), ('তোমায়', 'PN'), ('ভালোবাসি', 'VM'), ('|', 'PU')]

Chunked sentence-1: [('আমি', 'PN', 'B-NPP'), ('তোমায়', 'PN', 'I-NPP'), ('ভালোবাসি', 'VM', 'B-VPP'), ('|', 'PU', 'O')]

Tree representation of sentence-1:

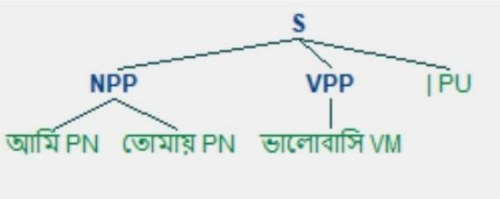

Figure 4.1: Partially parsed tree structure of sentence-1

Input sentence-2: আমার সোনার বাংলা , আমি তোমায় ভালোবাসি |

POS tagged sentence-2: [('আমার', 'PN'), ('সোনার', 'NC'), ('বাংলা', 'NP'), (',', 'PU'), ('আমি', 'PN'), ('তোমায়', 'PN'), ('ভালোবাসি', 'VM'), ('|', 'PU')]

Chunked sentence-2: [('আমার', 'PN', 'B-NPP'), ('সোনার', 'NC', 'I-NPP'), ('বাংলা', 'NP', 'I-NPP'), (',', 'PU', 'O'), ('আমি', 'PN', 'B-NPP'), ('তোমায়', 'PN', 'I-NPP'), ('ভালোবাসি ', 'VM', 'B-VPP'), ('|', 'PU', 'O')]

Tree representation of sentence-2:

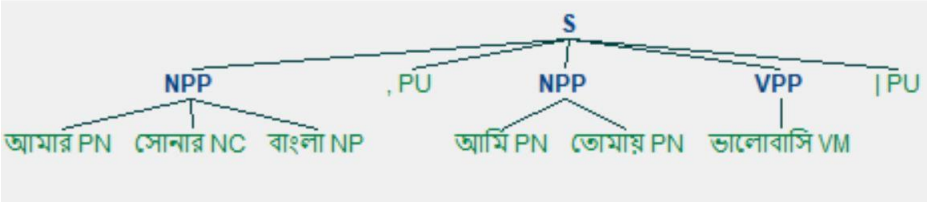

Figure 4.2: Partially parsed tree structure of sentence-2

Input sentence-3: মলয়ের কাছে সব শুনে তার স্ত্রী বলল , " তোমার নাম কি মা ? "

POS tagged sentence-3: [('মলয়ের', 'NP'), ('কাছে', 'PP'), ('সব', 'QT'), ('শুনে', 'VM'), ('তার', 'PN'), ('স্ত্রী', 'NC'), ('বলল', 'VM'), (',', 'PU'), ("", 'PU'), ('তোমার', 'PN'), ('নাম', 'NC'), ('কি', 'QQ'), ('মা', 'NC'), ('?', 'PU'), ("", 'PU')]

Chunked sentence-3: [('মলয়ের', 'NP', 'B-NPP'), ('কাছে', 'PP', 'O'), ('সব', 'QT', 'O'), ('শুনে', 'VM', 'B-VPP'), ('তার', 'PN', 'B-NPP'), ('স্ত্রী', 'NC', 'I-NPP'), ('বলল', 'VM', 'B-VPP'), (',', 'PU', 'O'), ("", 'PU', 'O'), ('তোমার', 'PN', 'B-NPP'), ('নাম', 'NC', 'I-NPP'), ('কি', 'QQ', 'O'), ('মা', 'NC', 'B-NPP'), ('?', 'PU', 'O'), ("", 'PU', 'O')]

Tree representation of sentence-3:

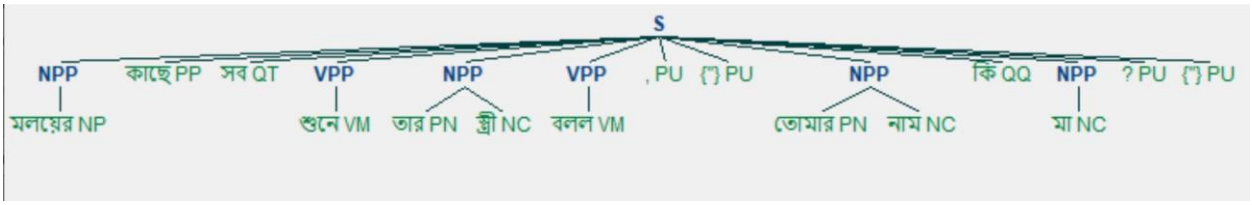

Figure 4.3: Partially parsed tree structure of sentence-3

Input sentence-4: আহা ! লোকটি দেখতে পায় না |

POS tagged sentence-4: [('আহা', 'IJ'), ('!', 'PU'), ('লোকটি', 'NC'), ('দেখতে', 'VM'), ('পায়', 'VA'), ('না', 'NG'), ('|', 'PU')]

Chunked sentence-4: [('আহা', 'IJ', 'O'), ('!', 'PU', 'O'), ('লোকটি', 'NC', 'B-NPP'), ('দেখতে', 'VM', 'B-VPP'), ('পায়', 'VA', 'I-VPP'), ('না', 'NG', 'O'), ('|', 'PU', 'O')]

Tree representation of sentence-4:

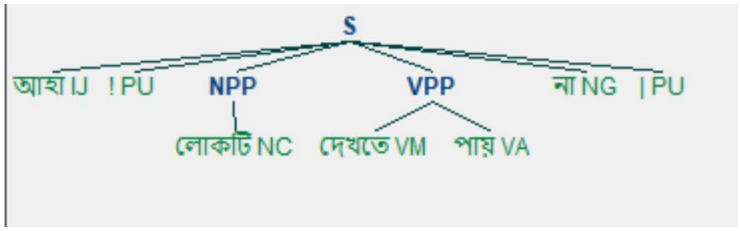

Figure 4.4: Partially parsed tree structure of sentence-4

Input sentence-5: তার শরীরে ভিটামিন A - এর অভাব আছে |

POS tagged sentence-5: [('তার', 'PN'), ('শরীরে', 'NC'), ('ভিটামিন', 'NC'), ('A', 'RD'), ('-', 'PU'), ('এর', 'PP'), ('অভাব', 'NC'), ('আছে', 'VM'), ('|', 'PU')]

Chunked sentence-5: [('তার', 'PN', 'B-NPP'), ('শরীরে', 'NC', 'I-NPP'), ('ভিটামিন', 'NC', 'I-NPP'), ('A', 'RD', 'O'), ('-', 'PU', 'O'), ('এর', 'PP', 'O'), ('অভাব', 'NC', 'B-NPP'), ('আছে', 'VM', 'B-VPP'), ('|', 'PU', 'O')]

Tree representation of sentence-5:

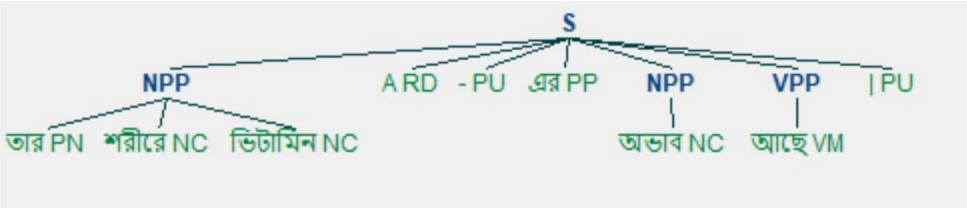

Figure 4.5: Partially parsed tree structure of sentence-5

Input sentence-6: রাজা মহানন্দ রাজধানীতে তৈরি করেছিল শিব মন্দির ও বৈষ্ণবদের মন্দির |

POS tagged sentence-6: [('রাজা', 'NP'), ('মহানন্দ', 'NP'), ('রাজধানীতে', 'NC'), ('তৈরি', 'VM'), ('করেছিল', 'VA'), ('শিব', 'NP'), ('মন্দির', 'NC'), ('ও', 'CJ'), ('বৈষ্ণবদের', 'NC'), ('মন্দির', 'NC'), ('|', 'PU')]

Chunked sentence-6: [('রাজা', 'NP', 'B-NPP'), ('মহানন্দ', 'NP', 'I-NPP'), ('রাজধানীতে', 'NC', 'I-NPP'), ('তৈরি', 'VM', 'B-VPP'), ('করেছিল', 'VA', 'I-VPP'), ('শিব', 'NP', 'B-NPP'), ('মন্দির', 'NC', 'I-NPP'), ('ও', 'CJ', 'O'), ('বৈষ্ণবদের', 'NC', 'B-NPP'), ('মন্দির', 'NC', 'I-NPP'), ('|', 'PU', 'O')]

Tree representation of sentence-6:

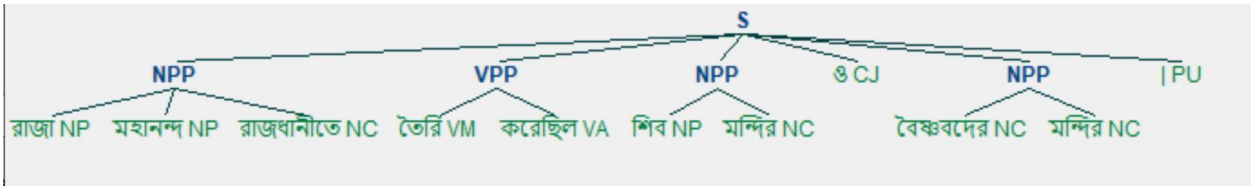

Figure 4.6: Partially parsed tree structure of sentence-6

Input sentence-7: আকবকের দরবারে যে ছত্রিশ জন বিশিষ্ট কলাকার ছিলেন , সকলেরই মতে , তাঁদের মধ্যে তানসেন ছিলেন সর্বশ্রেষ্ঠ |

POS tagged sentence-7: [('আকবকের', 'NP'), ('দরবারে', 'NC'), ('যে', 'PP'), ('ছত্রিশ', 'QT'), ('জন', 'NC'), ('বিশিষ্ট', 'AJ'), ('কলাকার', 'NC'), ('ছিলেন', 'VM'), (',', 'PU'), ('সকলেরই', 'PN'), ('মতে', 'NC'), (',', 'PU'), ('তাঁদের', 'PN'), ('মধ্যে', 'PP'), ('তানসেন', 'NP'), ('ছিলেন', 'VM'), ('সর্বশ্রেষ্ঠ', 'AJ'), ('|', 'PU')]

Chunked sentence-7: [('আকবকের', 'NP', 'B-NPP'), ('দরবারে', 'NC', 'I-NPP'), ('যে', 'PP', 'O'), ('ছত্রিশ', 'QT', 'O'), ('জন', 'NC', 'B-NPP'), ('বিশিষ্ট', 'AJ', 'O'), ('কলাকার', 'NC', 'B-NPP'), ('ছিলেন', 'VM', 'B-VPP'), (',', 'PU', 'O'), ('সকলেরই', 'PN', 'B-NPP'), ('মতে', 'NC', 'I-NPP'), (',', 'PU', 'O'), ('তাঁদের', 'PN', 'B-NPP'), ('মধ্যে', 'PP', 'O'), ('তানসেন', 'NP', 'B-NPP'), ('ছিলেন', 'VM', 'B-VPP'), ('সর্বশ্রেষ্ঠ', 'AJ', 'O'), ('|', 'PU', 'O')]

Tree representation of sentence-7:

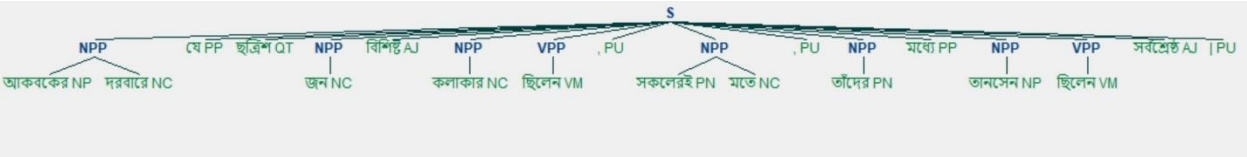

Figure 4.7: Partially parsed tree structure of sentence-7

Input sentence-8: মনে আছে , হেলিকপ্টারে চড়ে ঢুকেছিলাম জ্যান্ত আগ্নেয়গিরির গহ্বরে |

POS tagged sentence-8: [('মনে', 'NC'), ('আছে', 'VM'), (',', 'PU'), ('হেলিকপ্টারে', 'NC'), ('চড়ে', 'VM'), ('ঢুকেছিলাম', 'VM'), ('জ্যান্ত', 'AJ'), ('আগ্নেয়গিরির', 'NC'), ('গহ্বরে', 'NC'), ('|', 'PU')]

Chunked sentence-8: [('মনে', 'NC', 'B-NPP'), ('আছে', 'VM', 'B-VPP'), (',', 'PU', 'O'), ('হেলিকপ্টারে', 'NC', 'B-NPP'), ('চড়ে', 'VM', 'B-VPP'), ('ঢুকেছিলাম', 'VM', 'I-VPP'), ('জ্যান্ত', 'AJ', 'O'), ('আগ্নেয়গিরির', 'NC', 'B-NPP'), ('গহ্বরে', 'NC', 'I-NPP'), ('|', 'PU', 'O')]

Tree representation of sentence-8:

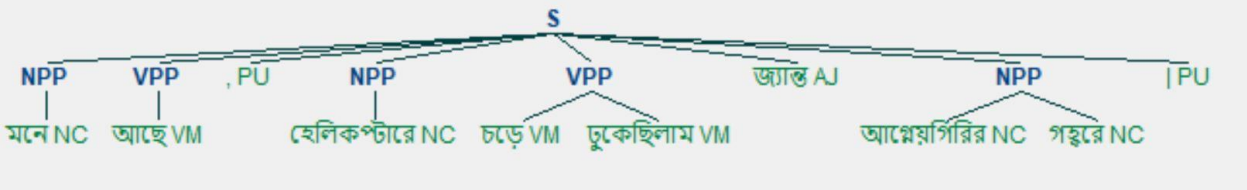

Figure 4.8: Partially parsed tree structure of sentence-8

Input sentence-9: ইহা খাদ্যকে ধরিতে এবং য়্যানটিনা , প্যাল্প এবং অগ্রপদকে পরিষ্কার রাখিতে সাহায্য করে |

POS tagged sentence-9: [('ইহা', 'PN'), ('খাদ্যকে', 'NC'), ('ধরিতে', 'VM'), ('এবং', 'CJ'), ('য়্যানটিনা', 'NC'), (',', 'PU'), ('প্যাল্প', 'NC'), ('এবং', 'CJ'), ('অগ্রপদকে', 'NC'), ('পরিষ্কার', 'AJ'), ('রাখিতে', 'VM'), ('সাহায্য', 'NC'), ('করে', 'VM'), ('|', 'PU')]

Chunked sentence-9: [('ইহা', 'PN', 'B-NPP'), ('খাদ্যকে', 'NC', 'I-NPP'), ('ধরিতে', 'VM', 'B-VPP'), ('এবং', 'CJ', 'O'), ('য়্যানটিনা', 'NC', 'B-NPP'), (',', 'PU', 'O'), ('প্যাল্ল', 'NC', 'B-NPP'), ('এবং', 'CJ', 'O'), ('অগ্রপদকে', 'NC', 'B-NPP'), ('পরিষ্কার', 'AJ', 'O'), ('রাখিতে', 'VM', 'B-VPP'), ('সাহায্য', 'NC', 'B-NPP'), ('করে', 'VM', 'B-VPP'), ('|', 'PU', 'O')]

Tree representation of sentence-9:

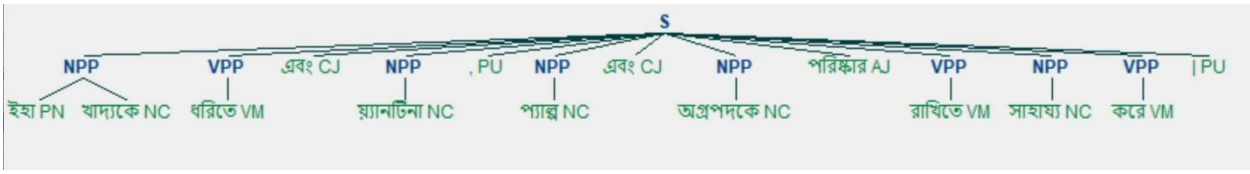

Figure 4.9: Partially parsed tree structure of sentence-9

Input sentence-10: বাবার শেষকৃত্য সম্পন্ন করেই সীমা ফিরে এসেছিল কলকাতায় |

POS tagged sentence-10: [('বাবার', 'NC'), ('শেষকৃত্য', 'NC'), ('সম্পন্ন', 'AJ'), ('করেই', 'VM'), ('সীমা', 'NP'), ('ফিরে', 'VM'), ('এসেছিল', 'VA'), ('কলকাতায়', 'NP'), ('|', 'PU')]

Chunked sentence-10: [('বাবার', 'NC', 'B-NPP'), ('শেষকৃত্য', 'NC', 'I-NPP'), ('সম্পন্ন', 'AJ', 'O'), ('করেই', 'VM', 'B-VPP'), ('সীমা', 'NP', 'B-NPP'), ('ফিরে', 'VM', 'B-VPP'), ('এসেছিল', 'VA', 'I-VPP'), ('কলকাতায়', 'NP', 'B-NPP'), ('|', 'PU', 'O')]

Tree representation of sentence-10:

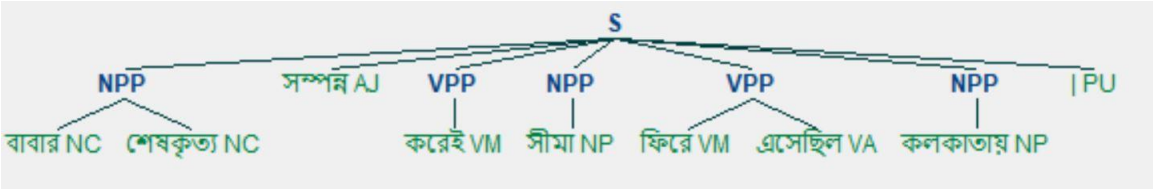

Figure 4.10: Partially parsed tree structure of sentence-10

4.1 Test Input Data

The test input data that can be processed through our shallow parser is illustrated bellow:

Serial no.	Test Input Sentences
01	আমি তোমায় ভালোবাসি।
02	আমার সোনার বাংলা , আমি তোমায় ভালোবাসি।
03	সততা একটি মহৎ গুণ।
04	সে পরিশ্রমী।
05	রহিম ভাত খাচ্ছে।
06	রহিম সৎ তাই সে মহৎ।

07	সে নিজে কাজটা করেছে।
08	আমাকে যেতে দাও।
09	সে দ্রুত দৌড়াতে পারে।
10	ছিঃ , এমন কাজ তোর।
11	আহা ! লোকটি দেখতে পায় না।
12	আমি তার জন্য অপেক্ষা করছি।
13	তুমি ও আমি যাব।
14	সে কি আসবে ?
15	তার শরীরে ভিটামিন A - এর অভাব আছে।
16	খুব কম ছেলেই ঘটকের চোখে পাত্রী দেখে বিয়ে করতে চায়।
17	রপ্তানি দ্রব্য - তাজা ও শুকনা ফল , আফিম , পশুচর্ম ও পশম এবং কার্পেট।
18	রাজা মহানন্দ রাজধানীতে তৈরি করেছিল শিব মন্দির ও বৈষ্ণবদের মন্দির।
19	মলয়ের কাছে সব শুনে তার স্ত্রী বলল , " তোমার নাম কি মা ? "
20	নিজস্ব প্রতিনিধি।
21	অন্ধকার গাঢ়তর হয়ে ওঠে।
22	এর দ্বারা বশীকরণও হয় , আচার্য্যেরা এই বলে থাকেন।
23	বিশ্বাস , ঠিকও হয়।
24	আকবকের দরবারে যে ছত্রিশ জন বিশিষ্ট কলাকার ছিলেন , সকলেরই মতে , তাঁদের মধ্যে তানসেন ছিলেন সর্বশ্রেষ্ঠ।

25	সাবিত্রীদি চুড়ো বেঁধে দিচ্ছিলেন সতুর চুলে।
26	মনে হয় , হতাশা লাঘব করার চেষ্টা।
27	বুধবার এঁদের মধ্যে একজনের মৃত্যু হয়।
28	তাহার জন্য তাহাদের দাম্ভিকতা নাই।
29	মনে আছে , হেলিকপ্টারে চড়ে ঢুকেছিলাম জ্যান্ত আগ্নেয়গিরির গহ্বরে।
30	সশব্দে ট্রাক ছুটছে।
31	নিতান্ত গৃহবধূ ছিলেন না কোরাজন আকিনো।
32	তোমার বন্ধু বলল দু হাজার হবে।
33	রাণী লক্ষ্মীবাই রাজসভা আহ্বান করেছেন।
34	ইহা খাদ্যকে ধরিতে এবং য়্যানটিনা , প্যাল্প এবং অগ্রপদকে পরিষ্কার রাখিতে সাহায্য করে।
35	ভাইকে প্রাণাধিক স্নেহ করতেন রামকুমার।
36	নির্মাণের কাজ চলছে মন্থর গতিতে।
37	টেবিল অনুনাদী বস্তুর কাজ করে।
38	বাবার শেষকৃত্য সম্পন্ন করেই সীমা ফিরে এসেছিল কলকাতায়।
39	প্রদর্শনী খেলা দুটি - ব্যাডমিণ্টন ও বোলিং।
40	সাধারণত নেতাজী ডিনারটা আনন্দ করে খেতেন।
41	আয়নাটির নাম ভালবাসা।
42	যা বেরিয়ে যা।

43	// বিংশ শতাব্দীর বিশের দশকে উচ্চ বিভব সম্পন্ন প্রতি প্রভব বাতি ব্যবহার হত।
44	রাজ্য শাখার এক বিবৃতিতে বলা হয়েছে , সি পি আই (এম) দেশের একটি স্বীকৃত রাজনৈতিক দল।
45	এর ফলে মরু অঞ্চলে তীব্র নিম্নচাপ সৃষ্টি হয়।
46	এরই নাম মায়া !
47	মালদহ জেলার একটি জায়গা থেকে রাজা শশাঙ্কের একটি স্বর্ণমুদ্রাও তাঁরা সংগ্রহ করতে পেরেছেন।
48	আগামী বছর 1990 সালকে বিশ্ব সাক্ষরতা দিবস হিসাবে পালন করা হবে।
49	শিল্প শিল্পীর অন্তরেই প্রকাশিত।
50	বিহারীলালের বিবাহ হয় 19 বছর বয়সে।
51	অকুস্থল চীনের একটি শহর।
52	তার বেঁচে থাকাই কঠিন হয়।
53	1939 সালে ব্রিটেনে চার হাজার টিভি সেট বিক্রি হয়।
54	মোজায় নেই ফুটো।
55	চিত্রতারকার বুশশার্টখানা মদনা সন্ধানী চোখে দেখে নিয়েছিল।
56	চার বছরে পঞ্চম মুদ্রণ ভ্রমণ সাহিত্যে সত্যিই অভাবনীয় এবং আনন্দের।
57	আরও বড় কিছু , বেশী কিছু।
58	সারা হায়দ্রাবদ শহরে উত্সবের হাওয়া।
59	শীতকাল বড় প্রবঞ্চক।

60	দুজনেই ইঙ্গিতটা ধরতে পেরেছেন।
61	বল্লারশাহ এবং ওয়ারধা থেকে য়্যামবুলেনস নিয়ে ছুটে গিয়েছেন চিকিৎসকরা।
62	যাঁহারা ধীশক্তিসম্পন্ন, সূক্ষ্মদর্শী, চিন্তাশীল কবি ও রসিক, তাঁহারাই প্রবাদের সৃষ্টিকর্তা।
63	একমাত্র শ্রমই পণ্য - মূল্য সৃষ্টি করে এবং শ্রমই মূল্যের মাপকাঠি।
64	বলতো ওদের ফ্ল্যাটের দামী দামী জিনিস চুরি হয়ে গেছে তোমার দোষে।
65	দুদিন সবুর করো মশাই।
66	ঈগল পাখি মাছ খায়।

Chapter 5

Conclusion

5.1 Conclusion

Shallow parsing is highly beneficial for different types of natural language processing applications like machine translation, relation extraction, information retrieval system where rarely requires complete parse tree generation. In this thesis work, we used maximum entropy modeling for parts-of-speech tagging which provides better accuracy in terms of other POS tagging methods. For general words, the previous two words and their tags and next two words and their tags according to the current word, for rare words four prefix and four suffix information are used as feature set so the feature extraction process is highly effective for solving syntactical ambiguities, and better POS tagging provides better accuracy during chunking process. Through our POS tagger we have achieved nearly 90.78% accuracy for POS tagging and after chunking POS tagged data, we achieved nearly 88% accuracy, the overall accuracy for shallow parsing is nearly 88%.

5.2 Future works

- We will add more POS tagged sentences to the train data to provide more accurate result.
- Currently our partial parser can accurately parse one test input at a time, in future paragraph can be setup to be partially parsed.
- More feature set can be added for feature extraction during POS tagging which can solve more ambiguities and dependencies.
- More chunking rule can be added if new train sentences are added with new syntactic rules.

References

[1] I. Ariaratnam, A. R. Weerasinghe, C Liyanage, "A Shallow parser for Tamil", International Conference on Advances in ICT for Emerging Regions (ICTer): 197 – 203, (2014, IEEE).

[2] Abney, S. (1991), "Parsing By Chunks". In: Robert Berwick, Steven Abney and Carol Tenny (eds.), Principle-Based Parsing. Kluwer Academic Publishers, Dordrecht, MA (pp 257-278).

[3] Asif Ekbal, Rejwanul Haque, Sivaji Bandyopadhyay, "Maximum Entropy Based Bengali Part of Speech Tagging", Advances in Natural Language Processing and Applications Research in Computing Science 33, pp. 67-78, 2008.

[4] Medari Tham,"Khasi Shallow Parser", Proc. of ICON-2018, NLPAI, pages 43–49.

[5] Sandipan Dandapat, "Part-of-Speech Tagging fo Bengali", In Proceedings of the SPSAL Workshop, IJCAI,2007, pp. 56-75.

[6] D. Jurafsky, J. H. Martin, A. Kehler, K. Vander Linden and N. Ward, "Speech and Language Processing: An Introduction to natural Language processing, computational linguistics and speech recognition", MIT Press 2000, vol. 2.

[7] Sivaji Bandyopadhyay and Asif Ekbal," HMM Based POS Tagger and Rule-Based Chunker for Bengali", Advances in Pattern Recognition, 2006, pp. 384-390.

[8] Al-Mahmud, Bishnu Sarker, K M Azharul Hasan, "Parsing Bangla Grammar Using Context Free Grammar (CFG)," In Technical Challenges and Design Issues in Bangla Language Processing, IGI Global, pp. 137-154, 2013.

[9] http://www.nltk.org/_modules/nltk/classify/maxent.html

[10] http://users.umiacs.umd.edu/~hal/megam/version0_3/