Deep Learning based Morphological Analysis for Bhojpuri

Report submitted in fulfillment of the requirements for the Exploratory Project of

Second Year B.Tech.

by

Ankit Kumar(19075009)
Sakshi Maheshwari(19075064)

Under the guidance of

Dr. Anil Kumar Singh

Department of Computer Science and Engineering
INDIAN INSTITUTE OF TECHNOLOGY (BHU) VARANASI
Varanasi 221005, India

May 2021

Dedicated to

Our teachers, parents, and the almighty God.....

Declaration

We certify that

1. The work contained in this report is original and has been done by ourselves and the general supervision of our supervisor.

2. The work has not been submitted for any project.

3. Whenever we have used materials (data, theoretical analysis, results) from other sources, we have given due credit to them by citing them in the text of the thesis and giving their details in the references.

4. Whenever we have quoted written materials from other sources, we have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

Place: IIT (BHU) Varanasi
Date: 12/05/2021

Ankit Kumar and Sakshi Maheshwari
B.Tech, B.Tech
Department of Computer Science and Engineering,
Indian Institute of Technology (BHU) Varanasi,
Varanasi, INDIA 221005.

<u>Certificate</u>

This is to certify that the work contained in this report entitled ***"Deep Learning based Morphological Analysis for Bhojpuri"*** *being submitted by* ***Ankit Kumar(19075009) and Sakshi Maheshwari(19075064)*** *, carried out in the Department of Computer Science and Engineering, Indian Institute of Technology (BHU) Varanasi, is a bona fide work of my supervision.*

Place: IIT (BHU) Varanasi
Date:12/05/2021

Dr. Anil Kumar Singh
Department of Computer Science and Engineering,
Indian Institute of Technology (BHU) Varanasi,
Varanasi, INDIA 221005.

Acknowledgments

We would like to express our sincere gratitude to our supervisor Dr. Anil Kumar Singh and our mentor Amit Kumar for their constant guidance and support during the whole project work.

Place: IIT (BHU) Varanasi **Ankit Kumar, Sakshi Maheshwari**

Date:12/05/2021

Abstract

Morphological analysis is an essential first step in tasks like machine translation and dependency parsing of morphologically rich languages (MRLs). However, the uncertainties introduced by the reunion of morphemes, constructing numerous possible inflections for a word, make the prediction complicated for MRLs.

So, we propose a context-based morphological analyzer, which analyses the words based on multitask learning of word-level tag markers for Bhojpuri. It predicts the complete set of morphological tags, Parts-of-speech (POS), Gender (G), Number (N), Person (P), Case (C), Tense-Aspect-Modality (TAM) marker, as well as the Lemma (L) for the words of the language.

Contents

List of Figures

List of Tables

List of Symbols

Symbol	Description
λ	Weight factors for Tags and Lemma
θ	Set of shared features
L_θ	Loss function over set of shared features

Chapter 1

Introduction

1.1 Overview

Morphological analysis refers to the task of assigning a set of well-defined morphological tags and a lemma (root) to the data of a language by studying various syntactic attributes such as inflection, derivation and combining forms.

In layman terms, it is a study of word formation, i.e. how words are built using smaller parts, and knowing those parts makes it easier to translate a word in one language into another.

In this, words are analyzed into their linguistic components and deal with the features associated with individual components. We consider six universal tags that characterize the morphology of every word in Indic languages: Parts-of-speech (POS), Gender (G), Number (N), Person (P), Case (C), Tense-Aspect-Modality (TAM), and the Lemma (L).

Some examples of features associated with a word are:

- Whether it is a noun, pronoun, adjective, adverb, verb, etc.
- Whether it is singular or plural.
- Whether it is masculine, feminine, or neutral, i.e., Gender.
- Whether it is present, past, or future tense, i.e., Tense.

For example:

- **Books**: Book + Noun + Plural.
- **Slept**: Sleep + Verb + Past tense

There are a few cases when a word may have multiple meanings depending upon the context for example, books in "He *books* a ticket" and in "He has a lot of *books*". So, in such cases, the context has to be considered as well, and such type of Morphological Analysis is known as *Context based Morphological Analysis*.

1.2 Need of the Research

Bhojpuri is an Indo-Aryan language mainly spoken in Bihar, Uttar Pradesh, some parts of Nepal, and in Mauritius (2^{nd} most spoken language). It is not an endangered language yet, though speakers are limited only to a small geographical region. We think this language needs more attention in the NLP fields because of its morphologically rich, non-configurational, and agglutinative nature. But we think that this language needs to evolve along the present technology trends. So, we are trying to take a step in its direction.

1.3 Motivation of the Research Work

Morphological analysis is an active research topic in the field of NLP as a pioneer to a range of complex tasks such as word-sense disambiguation, spell-checker, machine translation, and dependency parsing.

In machine translation of low resource languages, morphological analysis helps in sparsity reduction by breaking down each word into its tags and lemma. The translation framework then needs to translate only the lemma and then use the tags to re-generate the exact inflection of the word in the target language.

1.4 Organisation of the Report

The second chapter contains some related basic information of the terms and tools used throughout the report, quoted from relevant sources. Its first section has a brief explanation of some prerequisites. The transliteration scheme used in the project, i.e., the WX notation, is explained in the second section.

The third chapter describes the theoretical aspects of the project work. Its first section contains the description of the task. The second section has the details about the dataset and the source from where it is fetched. Next, we have highlighted the challenges faced during the research work. The last and most crucial section of this chapter describes the approach and methodology in detail.

The fourth chapter has the results and plots obtained from the model. In the first section of this chapter, we have compared the results, i.e., accuracy, precision, recall, etc., obtained from our model with the research paper. The second section has the plots and figures obtained from the model.

In the last chapter, we finally conclude and have written about the future aspects of this project.

Chapter 2

Background Knowledge

2.1 Some definitions

This section contains the basic definitions and explanation of some terminologies used in the report.

- **Morpheme** : "A morpheme is the smallest meaningful unit in a language. It is a word or a part of a word that has meaning. It cannot be divided into smaller meaningful segments without changing its meaning or leaving a meaningless remainder. It has relatively the same stable meaning in different verbal environments."[1]

- **Root** : "A root is a form which is not further analysable, either in terms of derivational or inflectional morphology. It is that part of word-form that remains when all inflectional and derivational affixes have been removed. A root is the basic part always present in a lexeme."[2]

- **Affix** : "An affix is a morpheme that is attached to a word stem to form a new word or word form."[3]

- **Inflection** : "Inflection is a process of word formation, in which a word is modified to express different grammatical categories such as tense, case, voice, aspect, person, number, gender, mood, animacy, and definiteness."[4]

- **Morphological tagging** : "Morphological tagging is the task of assigning labels to a sequence of tokens that describe them morphologically. Apart from Part-of-speech tags, morphological tagging also considers morphological features, such as case, gender, or the tense of verbs."[5]

  The tags that we are considering in the project are listed below:

  1. **Part of Speech** : POS tag of a word represents the particular part of speech, depending on the definition of the word and its context, for example POS tag of slept is Verb.
  2. **Gender** : Gender tag of a word states if the word has masculine tone to it, or feminine.
  3. **Number** : Number tag of a word states if the word is singular or plural.
  4. **Person** : Person tag of a word represents if it is in 1^{st} person, 2^{nd} person or 3^{rd} person.
  5. **Case** : Case tag of a word indicates the grammar function according to their relation with rest of the words in a sentence.
  6. **Tense-Aspect-Modality** : As the name suggests, this tag describes about 3 important points of a word: Tense(Past/Present/Future), Aspect(Perfective/Imperfective) and Mood(Realis/Irrealis).

2.2 WX Notation

WX notation is a transliteration scheme to represent a script in ASCII form or the Roman alphabet. WX notation defines a standard that provides a unique representation of the Indian languages into the Roman alphabet.

Each alphabet has a unique mapping into Roman script, i.e., it is a prefix code:

- For unaspirated consonants and short vowels, lowercase letters are used.
- For aspirated consonants and long vowels, uppercase letters are used.
- For retroflexed voiceless and voiced consonants, ‘t, T, d, and D’ are used.
- For dentals, ‘w, W, x, and X’ are used.

Figure 2.1 WX notation for Bhojpuri Language

Chapter 3

Project Work

3.1 Task

The task was to design a model to morphologically analyze each token of the Bhojpuri dataset and predict the Lemma (root word) and morphological tags such as Parts-of-speech (POS), Gender (G), Number (N), Person (P), Case (C), and Tense-Aspect-Modality (TAM).

3.2 Dataset

The dataset used in this project is fetched from the Universal Dependencies Treebank[6] and Hindi-Urdu Dependency Treebanks[7] hosted by IIIT-H. The treebank consists of sentences extracted from conversations and news articles.

The fact that the Hindi and Bhojpuri treebanks have words manually annotated with the correct morphological tags and the correct lemma in the contexts of the sentences the words occur, which helps us in context-based morphological analysis.

The format used in those Treebank is the default CoNLL format, in which the word is followed by the root and then all the morphological tags (separated by '—').

Example: 1 gaila jA v|gen-m|num-sg|per-any|case-|tam-|...

SNo	Word	Root	POS	Gender	Number	Person	Case	TAM	...
1	gaila	jA	v (Verb)	m (Male)	sg (Singular)	any (Any)	-	-	... ...

3.3 Challenges

Due to the unavailability of enough formatted data for NLP in Bhojpuri, we faced Zero Shot Condition and had to choose Hindi (in WX notation) as the training dataset and then we ran the available Bhojpuri dataset for testing.

3.4 Approach

We extended an exisiting analyzer named "Multi Task Deep Morphological Analyzer"[8]. We have used Unsupervised Domain Adaptation in Zero Shot Condition.

Domain adaptation[9] is the technique to implement an algorithm trained on one or more source data to a different but related target data where the source and target data have the same feature space but different distribution of features. It is a subcategory of transfer learning.

Let X be the source space (or description space D_s) and let Y be the target space (or label space D_t). Then the domain adaptation on $X \times Y$, trains the model $m : X \to Y$ on the source domain D_s such that it has minimum possible error on data from the target domain D_t.

Unsupervised domain adaptation (UDA) is the task of training a model on labeled data from the source domain to improve the model's performance on the unlabeled data from the target domain.

Zero-shot learning (ZSL)[10] is a problem setup in machine learning, where at test time, a learner observes samples from classes that were not observed during training, and now the model needs to predict the class they belong to. In simpler words, it is the situation obtained when we train a model on some classes and predict a new class, which the model has never seen before. It resembles a human's ability to generalize and identify new things without direct supervision. We have a pre-trained model, i.e., MT-DMA trained on the Hindi dataset, and then it serves as the basis for testing the Bhojpuri dataset. Zero-shot methods generally work by associating observed and non-observed classes through some form of auxiliary information, which encodes observable distinguishing properties of objects.

Since we are using MT-DMA as a base, which considers morphological analysis as a multi-task learning problem, it works on a set of shared features(θ) for the six morphological tags and lemma of a word by minimizing the cross-entropy loss function. Let, $i = 1, 2, ..., N$ be the individual observation instances, $c = 1, 2, ..., C$ the classes, and $j = 0, 1, 2, ..., 6$ the number of tasks, then the combined cross-entropy loss function of the model can be stated as:

$$L_\theta = \sum_{j=0}^{6} \lambda \Bigg(-\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} 1_{y_i \in C_c} log \Big(p_{model} \big[y_i \in C_c \big] \Big) \Bigg)$$

where, $p_{model}[y_i \in C_c]$ is a C-dimensional vector denoting the probability of i^{th} observation belonging to the c^{th} category

Further, the weight factors $\big(\lambda_0, \lambda_1, ..., \lambda_6\big)$ check the discounts imparted upon the individual loss of the tasks. We keep two sets of weight factors for the task, $\lambda = \big\{\lambda_{POS}, \lambda_{GEN}, ..., \lambda_{TAM}\big\}$ for the six tags and $\big\{\lambda_L\big\}$ for the lemma, and as a heuristic for searching a good approximation to these, we ensure that these sum up to 1.

Our system integrates two components: (a) the Tag Predictor and (b) the Lemma Predictor.This can be depicted from the figure given below :

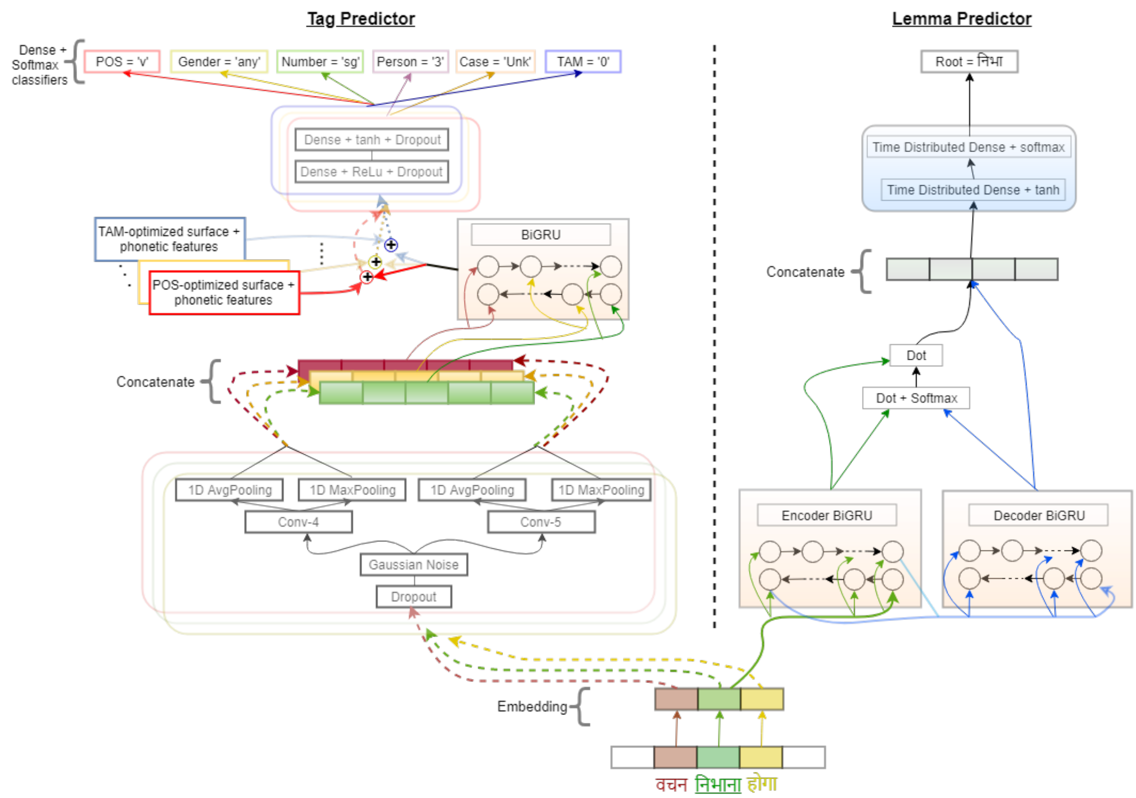

Figure 3.1 MT-DMA framework, with an example for morphological analysis of the word *Nibhaana*

The input to the model comprises a sequence of up to $2 * w + 1$ words, where w is the length of the context window (CW) of the target word. The final output consists of the predicted set of tags (POS, G, N, P, C, and TAM) yielded by the tag predictor and the Lemma (L) generated by the lemma predictor.

We employ character-level embeddings to capture intra-word morphological and shape information. The tag and the lemma predictors share a character embedding layer representing the characters of the input words using a 64-dimensional vector.

1. **The Tag Predictor:**

   The tag predictor employs two regularization techniques to prevent overfitting of the computed embeddings:

(a) A dropout[11] rate of 0.50 :

Dropout is a combo of learning, prediction, and regularization technique applied to deterministic feed-forward architectures that predict a target y given input vector v. These architectures contain a series of hidden layers h. Dropout trains an ensemble of models consisting of all models with a subset of the variables in both v and h.

Dropout is easily implemented by randomly selecting nodes to be dropped out with a given probability each weight update cycle. We add a Dropout layer between the input (or visible layer) and the hidden layer and set the dropout rate as 50%, meaning one in 2 inputs will be randomly excluded from each update cycle.

(b) A Gaussian noise layer[12] with a mean of zero :

For the tag predictor model, we found that its dependency upon the context words could make the model sensitive to the irregularities in the context, i.e., spaces, misplaced punctuation marks, and lack of enough context words in case of shorter sentences. To build resistance to such disturbances, we insert additive zero-centered Gaussian noise into the embedded character inputs for all words of the tag predictor. Gaussian Noise (GS) is a regularization layer acting as a corruption process for real-valued inputs.

The noise-injected vectors are fed into separate sets of convolution and pooling layers that help capture short-range regularities among the characters of individual words. A Bidirectional Gated Recurrent Unit (Bi-GRU)[13]layer combines the convolutions of the current word with those in its context to process the long-range dependencies among all such features. The representation obtained from the BiGRU is then branched off with each such branch merging separately into tag-specific hand-crafted features for each of the six tags, selected from a pool of linguistic features via a multi-objective GA optimization. Each branch is processed by six separate stacks of two dense and dropout layers before being passed onto a final dense layer with softmax activation, which serves to classify the tag's set of possible markers.

2. **The Lemma Predictor:**

   The lemma predictor is essentially a seq2seq variant of the encoder-decoder model that shares the same embedding space as the tag predictor to perform character-level transliteration of the word.

   The encoder feeds upon the character embeddings of the current word. Because of the sequential processing of the characters by the encoder, the state of the encoder at one-time step t_h serves to capture a summary of all the previous character sequences, i.e., t_1, t_2, ..., t_{h-1}.

   The decoder generates the root word by inputting the hidden states of the last time step of the encoder, thus capturing the summary of the entire character sequence.

   The lemma predictor model being context-independent does not require any noise injections. The lemma predictor leverages a Bi-GRU based encoder-decoder architecture, thus treating the problem as a sequence-to-sequence mapping between the characters of the input word and the lemma. Unlike the tag predictor, the input to the root predictor model comprises only the current word and not its context.

Chapter 4

Results and Plots

4.1 Results

The next step in the project is to check if the obtained results have similar accuracy as written in the paper(MTDMA). So, to evaluate the quality of our Model's predictions we used the following parameters:

1. **Accuracy:** Accuracy measures the fraction of correctly predicted values.
2. **Precision:** Precision can be defined as fraction of positively predicted values.It measures the quality of a model.
3. **Recall:** Recall is fraction of positively predicted values correctly predicted by the model.It measures the quantity of a model.
4. **F1-score:** F1 score is harmonic mean of precision and recall.It tells about how precise and robust is our predictions.

For calculating the parameters listed above, we have used a python package named **sklearn.metrics** by ScikitLearn.

Analysis	Accuracy (Official UTF)	Accuracy (Our model)	
		UTF	WX
Lemma	99.27	93.84	95.14
POS	99.66	98.09	95.99
Gender	99.33	95.33	94.98
Number	98.25	95.75	93.27
Person	98.93	97.03	94.54
Case	97.41	96.92	93.55
TAM	99.68	97.0	96.42

Table 4.1 Comparison between the accuracy written in the paper and the accuracy our model generates for Hindi dataset(averaged result after 10 runs)

Analysis	UTF	WX
Lemma	64.8	63.88
POS	82.54	77.26
Gender	31.57	45.54
Number	39.58	52.75
Person	43.33	56.77
Case	35.91	51.21
TAM	41.48	56.41

Table 4.2 Comparison between the accuracy using the UTF model and accuracy using the WX model on the Bhojpuri Testset

Hence, from the table above, we can conclude that WX version performs better than the UTF version in almost every row. Hence, it should be chosen.

4.1.1 Confidence Interval

Then, Confidence Interval is calculated for the model after 10 successful runs.

$$\text{Confidence Interval} = \overline{x} \pm z \cdot \frac{s}{\sqrt{n}}$$

where, $\overline{x}$ is the mean of the data,
z is the confidence level value,
s is the standard deviation of the data,
n is the total number of data,

S. No.	1	2	3	4	5	6	7	8	9	10
Accuracy	62.53	63.9	64.28	60.03	59.9	63.88	64.01	62.77	61.86	64.9

Table 4.3 The accuracy of our model over 10 runs(5 epochs each)

The values in our case are:

$\overline{x} = 62.806$

$s = 1.747$

$n = 10$

$z = 1.645$ (for 90% Confidence Level)

$$\text{Confidence Interval for accurcay} = 63.806 \pm 1.645 \cdot \frac{1.747}{\sqrt{10}}$$

Therefore, Confidence Interval for accuracy $= 62.806 \pm 0.909 \ \left(\pm 1.45\% \right)$

4.2 Plots

This section contains the various plots and figures generated from the model.

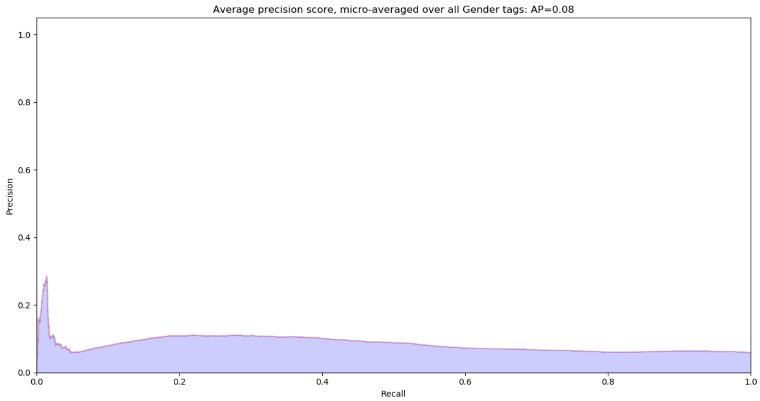

Figure 4.1 Average precision score, micro-averaged over all Gender tags, AP=0.8

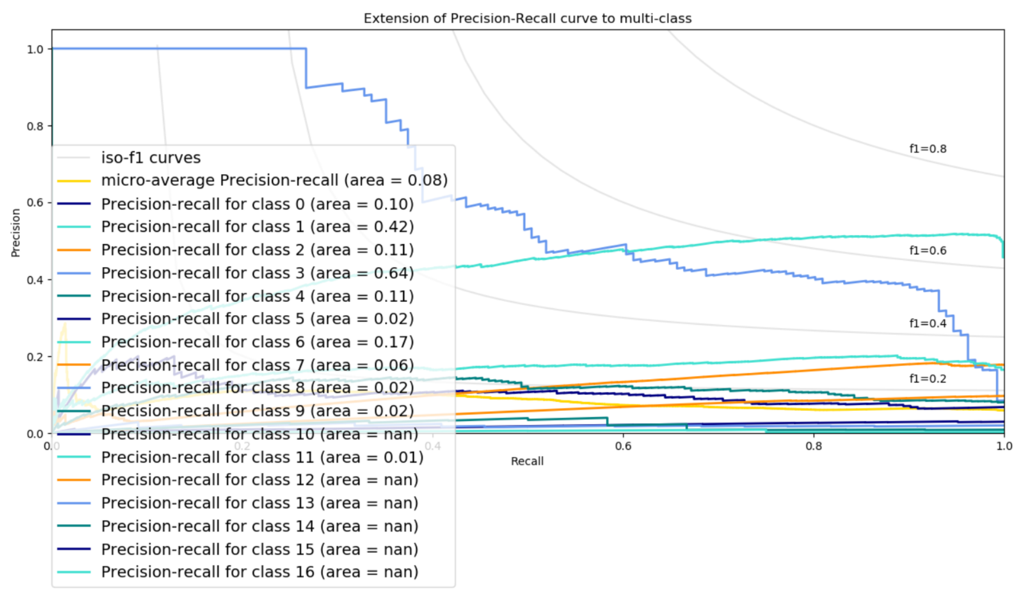

Figure 4.2 Extension of Precision-Recall curve to multi-class

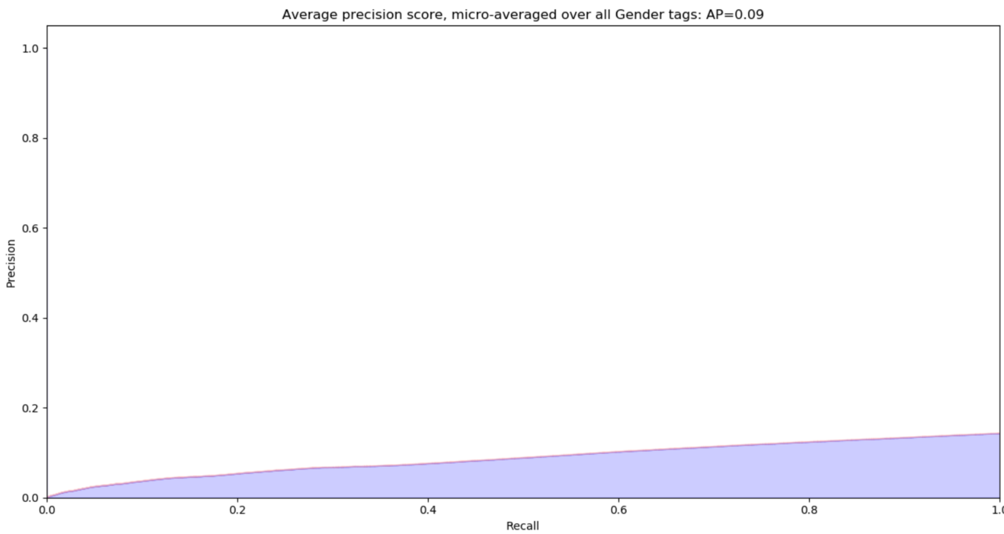

Figure 4.3 Average precision score, micro-averaged over all Gender tags, AP=0.9

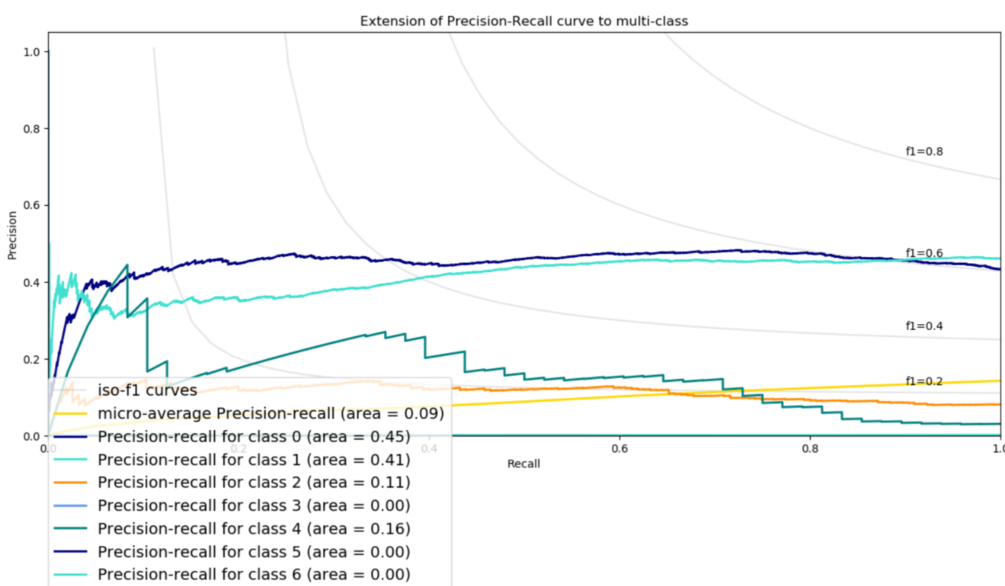

Figure 4.4 Extension of Precision-Recall curve to multi-class

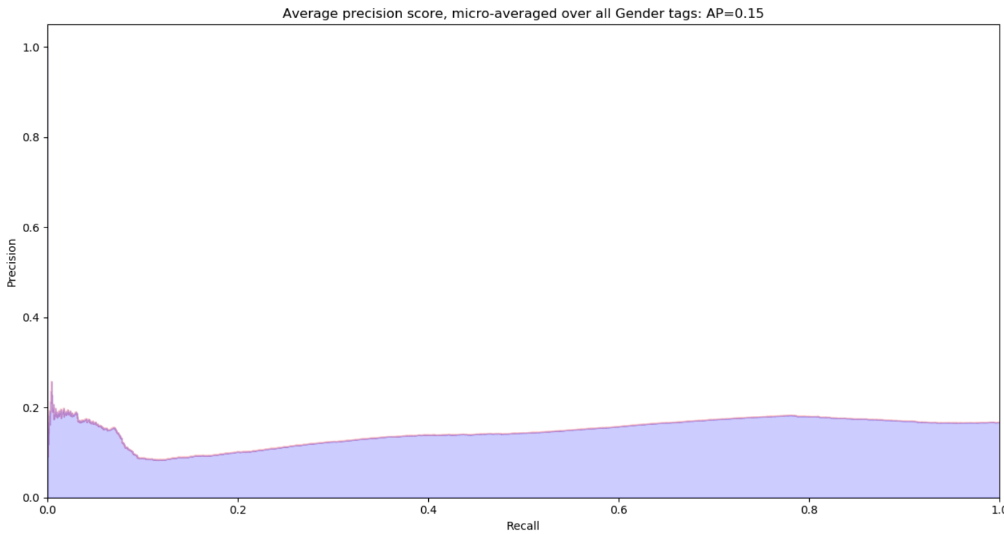

Figure 4.5 Average precision score, micro-averaged over all Gender tags, AP=0.15

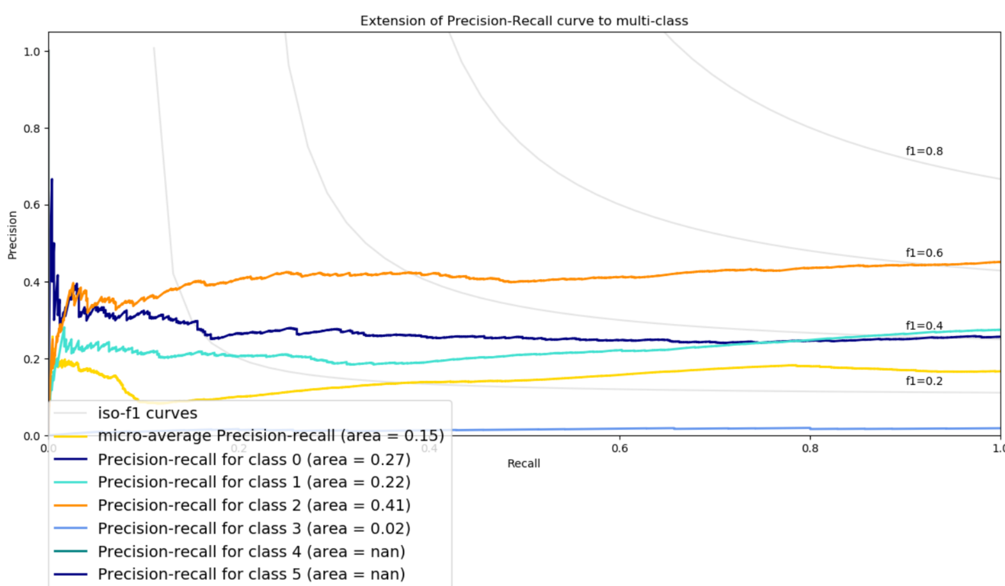

Figure 4.6 Extension of Precision-Recall curve to multi-class

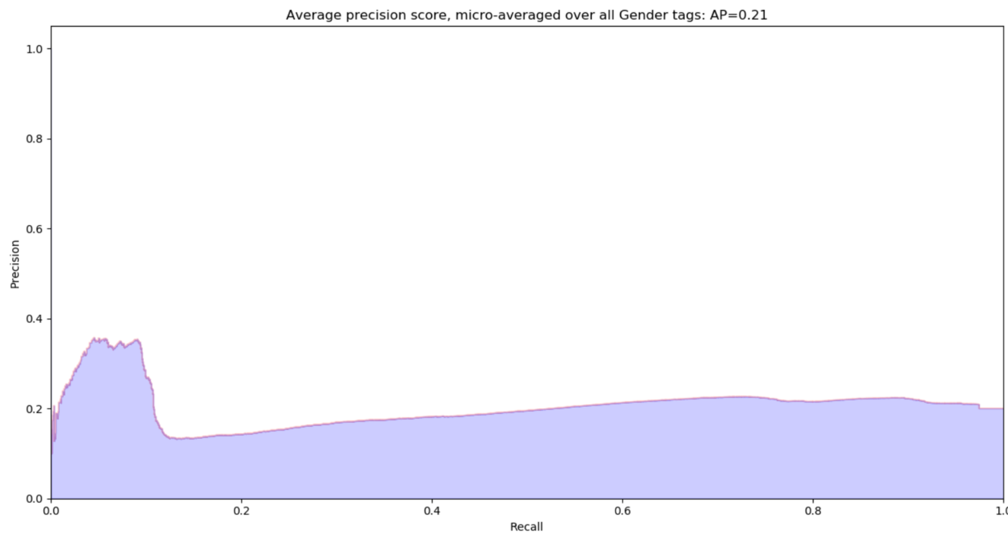

Figure 4.7 Average precision score, micro-averaged over all Gender tags, AP=0.21

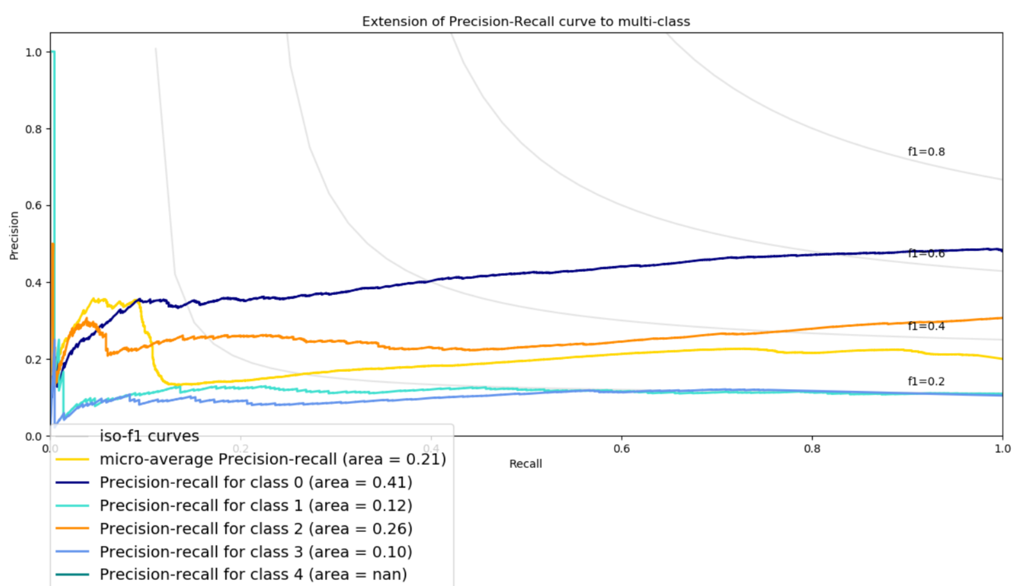

Figure 4.8 Extension of Precision-Recall curve to multi-class

Chapter 5

Conclusions and Discussion

- From our research, we conclude that using unsupervised domain adaptation in zero-shot learning condition on the MT-DMA, we can morphologically analyze any language, just like we did for Bhojpuri, even if the training dataset for the language is not available. This would make morphological analysis possible for low resource languages, such as Maithili, Sinhala, etc.
- Our overall results are better than all the previous results, which either used less data or some other technique.

Future Aspects

- Here we have used Hindi's training data to train the model, due to unavailability of enough well-formatted data in Bhojpuri, so in future there is scope to train the model on Bhojpuri training data, which will definitely improve the results.
- Here we have ran only 5 epochs per training dataset due to unavailabilty of good computational resources. So, in future, we can increase the number of epochs which will enhance the model.
- Since, there were not enough data available, fine-tuning couldn't be done. If

databank is improved, the model will improve its results as well.

- We look forward to see developers build applications like translator for Bhojpuri as well, after the enough training data is made.

Bibliography

[1] What are morphemes? [Online]. Available: https://www.rit.edu/ntid/sea/processes/wordknowledge/grammatical/whatare

[2] What are roots in nlp? [Online]. Available: https://www.researchgate.net/post/What-is-the-difference-between-root-word-and-stem-word

[3] Affix. [Online]. Available: https://en.wikipedia.org/wiki/Affix

[4] Inflection. [Online]. Available: https://en.wikipedia.org/wiki/Inflection

[5] Morphological tagging. [Online]. Available: https://paperswithcode.com/task/morphological-tagging

[6] Universal dependencies. [Online]. Available: https://github.com/UniversalDependencies/UD_Bhojpuri-BHTB/tree/master

[7] Hindi urdu dependency treebanks by iiit h. [Online]. Available: http://ltrc.iiit.ac.in/hutb_release/

[8] S. Jha, A. Sudhakar, and A. K. Singh, "Multi task deep morphological analyzer: Context aware joint morphological tagging and lemma prediction," *arXiv preprint arXiv:1811.08619*, 2018.

[9] Domain adaptation. [Online]. Available: https://en.wikipedia.org/wiki/Domain-adaptation

[10] Zero shot learning. [Online]. Available: https://en.wikipedia.org/wiki/Zero-shot-learning

[11] Dropout. [Online]. Available: https://www.researchgate.net/publication/259440611_An_empiri

[12] Gaussian noise layer. [Online]. Available: https://core.ac.uk/download/pdf/162017899.pdf

[13] Bidirectional gated recurrent unit. [Online]. Available: https://medium.com/analytics-vidhya/bi-directional-rnn-basics-of-lstm-and-gru-e114aa4779bb