BENGALI NAMED ENTITY RECOGNITION USING DEEP LEARNING

BY

Khadija Akter Lima
ID: 172-15-9651
AND

Md. Asadujjaman
ID: 172-15-10212

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Dr. Sheak Rashed Haider Noori
Associate Professor & Associate Head
Department of CSE
Daffodil International University

Co-Supervised By

Md. Zahid Hasan
Assistant Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

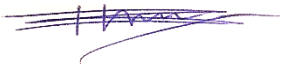
DHAKA, BANGLADESH

MAY 2021

APPROVAL

This Project titled "**Bengali Named Entity Recognition Using Deep Learning**", submitted by Khadija Akter Lima and Md. Asadujjaman to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on May 5, 2021.

BOARD OF EXAMINERS



Dr. Touhid Bhuiyan **Chairman**
Professor and Head
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Nazmun Nessa Moon **Internal Examiner**
Assistant Professor
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

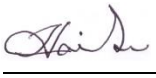
Aniruddha Rakshit **Internal Examiner**
Lecturer (Senior Scale)
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Dr. Mohammad Shorif Uddin **External Examiner**
Professor
Department of CSE
Jahangirnagar University

DECLARATION

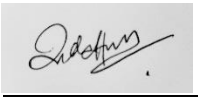
We hereby declare that, this project has been done by us under the supervision of **Dr. Sheak Rashed Haider Noori, Associate Professor & Associate Head, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Dr. Sheak Rashed Haider Noori
Associate Professor & Associate Head
Department of CSE
Daffodil International University

Co-Supervised by:



Md. Zahid Hasan
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:

Khadija Akter Lima
ID: 172-15-9651
Department of CSE
Daffodil International University

Md. Asadujjaman
ID: 172-15-10212
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to Almighty **Allah** for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Dr. Sheak Rashed Haider Noori**, Associate Professor & Associate Head, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Natural Language Processing*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. Touhid Bhuiyan**, **Professor and Head**, and Head**,** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

Named Entity Recognition (NER) is considered fundamental for extracting information in Natural Language Processing (NLP), and this task aims to classify each word of a text document into a list of predefined named entity classes. Numerous architectures for high-resource languages with high exactness, such as English and Chinese, have been built over time. In recent years, the NER challenge for low-resource languages like Bangla has piqued researchers' interest. To perform the NER task in low resource language Bangla, this work proposes a novel neural network that reduces the need for most feature engineering and aspires to utilize minimal information to get optimal performance. In this research, we have used a new dataset to observe various deep learning models' performance in respect of non-contextual word embedding such as word2vec, glove, and fastText. Consequently, a hybrid architecture made out of bidirectional Gated Recurrent Unit (BGRU), Convolutional Neural Network (CNN), and Conditional Random Field (CRF) emerged triumphant with the F1 Macro Score of 91.90%, and F1 Micro Score of 98.21%. Since precision, recall, and F1 were measured differently in different studies, this value may change. All of the experimental models have also been subjected to a previously introduced method for measuring precision, recall, and F1, with the proposed model scoring 86.83% on F1. The proposed BGRU-CNN-CRF architecture provides peak performance for all the non-contextual word embedding specified and has the highest accuracy for the word2vec word embedding. In addition, this study demonstrates the impact of a well-annotated dataset on accuracy by creating a unique dataset.

TABLE OF CONTENTS

LIST OF FIGURES PAGE

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Named Entity Recognition is the task of labeling each word of a sequence in certain predefined classes according to the context of that sequence. In Bangla NER, researchers have previously worked with different entities such as person, location, organization, date, time, unit, object, etc. In our research, we looked at six types of entities- person, location, organization, quantity, percentage, and currency. In our dataset, we tag only the individual's name as a person entity avoiding the designation of people. The name of any country or the name of any place within any country is tagged as a location entity. The organization entity includes all types of charities, educational institutions, business organizations, government, and non-government organizations. Money and percentage-related phrases are labeled as currency and percentage entities, respectively. Mass nouns, unit nouns, ordinal numbers, definite and indefinite numbers are under the quantity tag. Apart from these, we keep everything else in the other class. We have developed several hybrid models with Deep Learning and compared their performance utilizing three non-contextual word embedding.

1.2 Motivation

Bangla language has bought with blood. The bloodstains of February 21st, 1952 have not dried yet. There is no unfortunate nation in the world that has sacrificed its citizens for the sake of their mother tongue. We will never be able to repay this debt to the language martyrs. Other language speakers are doing multifaceted work with their language. We want to make this sweet language intelligent and take it to the understanding form of the machine. As a result, Bangla will have a significant presence in areas such as Deep Learning. We want to develop a quality dataset for the Bangla NER along with a robust system so that researchers can do multifaceted work with the Bangla language in the present and future.

1.3 Objective

Our main goal is to gain state-of-the-art performance for Named Entity Recognition in Bangla so that machine can work at a human level accuracy. We have created an enormous dataset so that researchers can use it for natural language processing research in Bangla. We are trying to develop a robust system that researchers can use for information extraction, anaphora resolution, machine translation, question answering, text summarization, etc. Our proposed system utilizes neural networks, rendering an optimal solution using minimal resources.

1.4 Research Questions

- Why do we need to create a dataset?
- What are the entities we need to work with?
- What amount of data should we collect?
- From where do we collect the data?
- Why should we use deep learning?
- What type of word embedding should we use?
- Why do we use character embedding?
- How do we extract the character level feature?
- Should we use LSTM or GRU?
- How to do performance analysis on the label imbalance dataset?

1.5 Expected Outcome

Our model will render the maximum accuracy using minimum resources. That is, the amount of error can be reduced as much as possible so that the performance of the machine is not reduced in any way. For this, we have to do it very carefully, from the dataset preparation to the model building. The system can be used to other sequence leveling tasks of NLP.

CHAPTER 2

BACKGROUND STUDY

2.1 Introduction

A lot of research has been done using named entity recognition in a variety of languages. High-resource languages such as English and Chinese have significantly improved accuracy and performance in the area of Named Entity Recognition. The first research work on Bangla NER began in West Bengal. The hand crafted feature was the focus of early stage research. Bangla language resources are significantly less than other languages. Researchers are currently trying to develop Named Entity Recognition for Bangla using minimal resources. With the advancement of Deep Learning technology, now it is possible to develop more efficient NER system that delivers human level accuracy. In this section, we discuss some renowned works on NER in different languages.

2.2 Related Works

Ekbal et al. [1] used a statistical Hidden Markov Model based NER system in both Bangla and Hindi language. The data source was the Bangla newspaper, which contains around 34-million-word forms. This study used suffix features and lexicon. Bangla word forms were 150,000 and Hindi 27,151 with 10-fold cross-validation. Besides, an HMM-based POS tagger was used to boost accuracy. For Bangla language Recall, Precision and F-score were reported as 90.2%, 79.48%, and 84.5%, while for Hindi language Recall, Precision, and F-score were reported as 82.5%, 74.6%, and 78.35% respectively.

Ekbal et al. [2] Ekbal used Support Vector Machine to build a NER scheme for the Bangla language. The Bangla newspaper served as the database. A total of 16 NE tags were used. The total amount of data was 150K, of which 130K was training data and 20K was test data, with 10-fold cross-validation. Recall, Precision, and F-score averages were 94.3 percent, 89.4 percent, and 91.8 percent, respectively.

Ekbal et al. [3] described a previous corpus of a Bangla newspaper with Named Entity tags and found a better result than previous studies that improved by 6.2% F-score using statistical Conditional Random Field. There are 17 name entity tags, divided into three major categories. Recall, Precision, and F-score were 93.8%, 87.8%, and 90.7% respectively.

Hasanuzzaman et al. [4] used the Maximum Entropy framework in both the Bangla and Hindi languages. The data source was IJCNLP-08 NER shared task on South and South East Asian Languages. There were four different types of tags. The train and test data for Bangla were 102,467 and 29K with 10-fold cross-validation respectively, while for Hindi they were 452,974 and 50K. The maximum entropy and marked Recall, Precision, and F-score for Bangla were 88.01%, 82.63%, 85.22%, while for Hindi they were 86.4%, 79.23%, 82.66% respectively.

Ekbal et al. [5] introduced a multi-engine approach in the Bengali language. The previous three models (SVM, CRF, and ME) were added in this Multi-Engine. The data source was a newspaper and NERSSEA. There were 17 tags with 4 categories. The training dataset was 272K words and the test data was 35K gold standard sentences. The lexical function was collected semi-automatically from unlabeled 3-million-word types. Recall, Precision and F-score were 93.98%, 90.63%, and 92.28% respectively

Parvez et al. [6] method was Hidden Markov Model on NER. Bangla news data from Ittefaq, BDNews, Prothom Alo, and other local newspapers were used as the data source. The number of tags was 4, and the total number of words was 56,196. Train data was 1 sentence with 21 tagged words and test data was 2 sentences. Recall, Precision and F-score were 1, 0.7, and 0.82 respectively.

Ibtehaz et al. [7] developed a partial string matching technique based on Breadth First Search on a Trie data structure which they combined with dynamic programming in Bangla. Unstructured textual data from Bangla newspapers, mostly from the Daily Prothom Alo (sports news), and structured data by using a Chabot to run structured queries were the data sources. The major tags used country names, player names, numbers.

Banik et al. [8] implemented the Gated Recurrent Unit based NER system in Bangla. 420 articles were collected from a reputed online Bangla newspaper. 4 major categories along with 3400 total tags have been used. For the limited dataset, the system provides an F1-score of 69%.

Chowdhury et al. [9] used a combination of various features with a CRF model. Bangla Content Annotation Bank was used as the data source. The number of tags was 7. Train data contained 24377 tokens from 1510 sentences and test data contained 6546 tokens from 427 sentences. Recall, Precision and F-score were 0.67, 0.78, and 0.72 respectively.

Rifat et al. [10] implemented a model using Bidirectional Gated Recurrent Unit with CNN in Bangla. The data source was a renowned newspaper. There were 8 numbers of tags with the total amount of tokens 96697. The amount of training token was 67554, while for testing it was 29143. Recall, Precision and F-score were 73.32%, 72.27%, and 72.66% respectively.

Karim et al. [11] proposed a NER system that combines the Densely Connected Network with the Bidirectional LSTM in Bangla. The data source was Bangla online news sources and Bangla Wikipedia. The total number of tokens was 983,663 from 71,284 sentences. Recall, Precision and F-score were 58.62%, 68.95%, and 63.37% respectively.

Ashrafi et al. [12] proposed a model which combines BERT, BLSTM, CRF, CW. The data source was Bangla online news sources and Bangla Wikipedia. The total nine numbers of tags used here. The total number of tokens was 98,85,090 from 72000 sentences. The training dataset contained 90% of the total data. The model achieved an F1 macro score of 65:96%, an F1 micro score of 90.64%, and an F1 Message Understanding Co-reference score of 72.04%.

Zhang et al. [13] showed LSTM outperforms with both word-based and character-based NER in Chinese. The lattice method is completely independent of word separation. There are 4 datasets-OntoNotes4, MSRA, Weibo NER, resume. The total eight number of tags used in this case. For OneNotes4 the Recall, Precision, and F-score were 76.35%, 71.56%, and 73.88% respectively. For MSRA, the Recall, Precision, and F-score were 93.57%, 92.79%, and 93.18% respectively. For Weibo NER, the Recall, Precision, and F-

score were 53.04%, 62.25%, and 58.79% respectively. For the resume dataset, the Recall, Precision, and F-score were 94.81%, 94.11%, and 94.46% respectively.

Li et al. [14] proposed machine reading comprehension as a unified framework that is for both flat and nested NER tasks in English and Chinese language. For the nested Named Entity Recognition, the Recall were 86.32%, 86.59%, 81.12%, and 77.61% on ACE04, ACE05, GENIA, and KBP-2017 dataset respectively. For the flat NER, the Recall was 94.61%, 89.95%, 95.12%, and 81.25% on English CoNLL 2003, English OntoNotes 5.0, Chinese MSRA, and Chinese OntoNotes 4.0 dataset respectively. For the nested Named Entity Recognition, the Precision was 85.05%, 87.16%, 85.18%, and 82.33% on ACE04, ACE05, GENIA, and KBP-2017 dataset respectively. For the flat NER, the Precision was 92.33%, 92.98%, 96.18%, and 82.98% on English CoNLL 2003, English OntoNotes 5.0, Chinese MSRA, and Chinese OntoNotes 4.0 dataset respectively. For the nested Named Entity Recognition, the F-score was 85.98%, 86.88%, 83.75%, and 80.97% on ACE04, ACE05, GENIA, and KBP-2017 dataset respectively. For the flat NER, the F-score was 93.04%, 91.11%, 95.75%, and 82.11% on English CoNLL 2003, English OntoNotes 5.0, Chinese MSRA, and Chinese OntoNotes 4.0 dataset respectively.

Chiu et al. [15] proposed a hybrid bidirectional LSTM and CNN architecture for the English NER. CoNLL-2003 and OntoNotes 5.0 datasets were used in this case study. The CoNLL-2003 dataset has 4 tags, and the OntoNotes 5.0 dataset has 18 tags. The number of tokens in train and test data was 204,567 and 46,666 for the CoNLL-2003 dataset whereas 1,088,503 and 152,728 for the OntoNotes 5.0 dataset. For both datasets, the F-score was 91.62% and 86.28% respectively.

Yang et al. [16] proposed a neural re-ranking system with LSTM and CNN structures for the English NER. CoNLL 2003 English dataset was used in this study. The number of tags was 4. The number of tokens on the train and test data were 14,987 and 3,684. The F-score was 91.62%.

Aguilar et al. [17] presented a multi-task neural network with CNN, BLSTM, and CRF for English NER on social media data. The Recall in entity and surface was 32.90% and 31.31% respectively. The Precision in entity and surface was 57.54% and 56.31%

respectively. The F-score in both entity and surface was 41.86% and 40.24% respectively.

Cao et al. [18] worked on an adversarial transfer learning framework with BLSTM and CRF for the Chinese NER. 2 datasets were used in this case study. The datasets were the Weibo NER dataset and Sighan2006 NER dataset. Training and testing data in the Weibo NER dataset were 1350 and 270, and for the SighanNER dataset were 41728 and 4365. The Recall, Precision, and F-score in SighanNER were 89.58%, 91.73%, and 90.64% whereas for the WeiboNER dataset were 50.68%, 55.72%, and 53.08% respectively.

Ghaddar et al. [19] proposed a vanilla recurrent neural network model (LSTM-CRF) for the Named Entity Recognition. Data was collected from Wikipedia ONTONOTES 5.0 and CONLL-2003. The total number of tags was 6. The amount of train data in CONLL-2003 was 204,567 whereas in ONTONOTES 5.0 was 1,088,503. The amount of test data in CONLL-2003 was 1,088,503 whereas in ONTONOTES 5.0 was 152,728. The F-score was 87.95% for the ONTONOTES 5.0 and 91.73% for the CONLL-2003.

Baevski et al. [20] designed a pretraining architecture based on a Bidirectional transformer for the English NER. The data sources were 3 types- Common Crawl, News Crawl, and BooksCorpus + Wikipedia dataset. The Common Crawl dataset was various subsets of Common Crawl which was web data. The train and test data sizes were 18B and 9B respectively. The News Crawl dataset was English news web data and the training size was 4.5B. The last dataset was collected from the BooksCorpus + Wikipedia with 800M words and English Wikipedia data of 2.5B words.

Jie et al. [21] proposed an effective dependency-guided LSTM-CRF model in English and Chinese NER. And the data source uses OntoNotes 5.0 dataset. The total of 18 tags were used here. For English, the Recall, Precision, and F-Score were 90.17%, 89.59%, and 89.88% respectively. For Chinese, the Recall, Precision, and F-Score were 78.86%, 81.00%, and 79.92% respectively.

Chen et al. [22] proposed a CNN based network for named entity recognition in the English language. The data source was CoNLL-2003 English NER, and Ontonotes 5.0 datasets, and the number of tags was 4. In the CoNLL-2003 English dataset, train and test

data contained 14,987 and 3,684 sentences respectively. In Ontonotes 5.0 datasets, train and test data contained 59,924 and 8,262 sentences respectively. The F-score for the CoNLL-2003 dataset was 91.44% whereas for the Ontonotes 5.0 dataset, 87.67%.

Jiang et al. [23] proposed differentiable neural architecture search methods for language modeling and Named Entity Recognition in the Chinese language. The data source was PTB corpus CoNLL-2003 and the F-score was 93.47%.

Strakova et al. [24] proposed two models using Neural Networks for the nested NER. The first one is LSTM-CRF architecture, and the 2nd one is a sequence-to-sequence task. The data sources were English ACE-2004, English ACE-2005, English GENIA, and Czech CNEC. The number of tags was 10. Nested NER results (F1) for the dataset ACE-2004, ACE-2005, GENIA and CNEC 1.0 (Czech) corpora were 84.40%, 84.33%, 78.31% and 86.88% respectively. Flat NER results (F1) for the dataset ACE-2004, ACE-2005, GENIA and CNEC 1.0 (Czech) corpora were 92.72%, 79.89%, 87.42% and 86.34% respectively.

Akbik et al. [25] proposed Pooled contextualized embedding for Named Entity Recognition in English, German and Dutch languages. The data sources were CONLL-03 (English, German and Dutch) and WNUT. The F-Score for the CONLL-03 dataset in English was 93.18%, German was 88.27%, and Dutch was90.44%. And the F-Score for the WNUT_17 dataset was 49.59%.

Yan et al. [26] proposed a NER architecture adopting adapted Transformer Encoder to model the character-level features and word level-features for Named Entity Recognition in English and Chinese language. English data sources were CoNLL2003, OntoNotes 5.0, and Chinese data sources were OntoNotes 4.0, MSRA, Weibo, and Resume. Train data for the CoNLL2003 dataset was 203.6k, OntoNotes 5.0 was 1088.5k, OntoNotes 4.0 was 491.9k, MSRA was 2169.9k, Weibo was 73.5k, and Resume was 124.1k. Test data for CoNLL2003 was 46.4k, OntoNotes 5.0 was 152.7k, MSRA was 172.6k, Weibo was 14.8k and Resume was 15.1k. The F-Score was 58.17 ± 0.2 for Weibo, 95.00 ± 0.25 for Resume, 72.43 ± 0.39 for OntoNotes4.0**,** 92.74 ± 0.27 for MSRA. The F-Score was in the

English language dataset 91.45 ± 0.07 for CoNLL2003 and 88.43 ± 0.12 for OntoNotes 5.0.

Li et al. [27] proposed a dice loss in replacement of the standard cross-entropy objective for data-imbalanced NLP tasks using Bidirectional Encoder Representations from Transformers, Machine reading comprehension, and Dice similarity coefficient in Chinese and English language. The English data source was CoNLL03, OntoNotes5.0 and the Chinese data source is MSRA and OntoNotes4.0. In this research, four tasks were evaluated- parts of speech tagging, named entity recognition, machine reading comprehension and paraphrase identification. The Recall, Precision, and F-Score for the CONLL-03 dataset in English were 93.41%, 93.25% and 93.33% respectively. The Recall, Precision, and F-Score for OntoNotes5.0 dataset in English were 91.59%, 92.56% and 92.07% respectively. The Recall, Precision, and F-Score for MSRA dataset in Chinese were 96.67%, 96.77% and 96.72% respectively. The Recall, Precision, and F-Score for OntoNotes4.0 dataset in Chinese were 84.22%, 84.72% and 84.47% respectively.

2.3 Comparative Analysis and Summary

A lot of work has been done in NER in the past. Many researchers have worked in different areas at different times throughout the NER. Various individuals show a variety of strategies for improving their performance. But a few renowned work has done for Bangla Named Entity Recognition. The previous work of Bangla NER is summarized in Table 2.1.

TABLE 2.1: A COMPARATIVE ANALYSIS OF DIFFERENT APPROACHES FOR NER IN BANGLA

Study	Method	Area	Tag Scheme& Tag Set	Corpus Source	Train & Test Data	Evolution (P-Precision R- Recall)
Ekbal et al. [1] Year: 2007	HMM	Person Location Organization Miscellaneous	Scheme: N/A Tag Set: 17	Newspaper	Train Data: 150000 (token) Test Data: 10-fold cross validation	P-79.52% R-90.30% F1-84.5%
Ekbal et al. [2] Year: 2008	SVM	Person Location Organization Miscellaneous	Scheme: N/A Tag Set: 17	Newspaper	Train Data: 150K (token) Test Data: 10-fold cross validation	P-89.4% R-94.3% F1-91.8%
Ekbal et al. [3] Year:2008	CRF	Person Location Organization Miscellaneous	Scheme: N/A Tag Set: 17	Newspaper	Train Data: 150K (token) Test Data: 10-fold Cross validation	P-87.8% R-93.8% F1-90.7%
Hasanuzzaman et al. [4]	ME	Person Location	Scheme: N/A	IJCNLP-08 shared task	Train Data:	P-82.63% R-88.01%

Year:2009		Organization Miscellaneous	Tag Set: 17	data	122,467 (token) Test Data: 10-fold Cross validation	F1-85.22%
Ekbal et al. [5] Year:2009	Voted System	Person Location Organization Miscellaneous	Scheme: N/A Tag Set: 17	IJCNLP-08 shared task data	Train Data: 272k (token) Test Data: 10-fold cross validation	P-90.63% R-93.98% F1-92.28%
Parvez et al. [6] Year: 2017	HMM	Person Location Organization Date Time	Scheme: N/A Tag Set: N/A	Newspaper	Train Data: 1 (sentence) Test Data: 2 (sentence)	P-85.07% R-94.07% F1-90%
Banik et al. [8] Year: 2018	GRU	Person Organization Location Day	Scheme: IOB Tag Set: 8	Newspaper	Train Data: N/A Test Data: N/A	F1- 69%
Chowdhury et al. [9] Year: 2018	LSTM + CRF	Person Location Organization Facility	Scheme: IOB2 Tag Set: 15	B-CAB	Train Data: 24377 (token)	P-0.65% R-0.53% F-0.58%

		Time Units			Test Data: 6546 (token)	
Karim et al. [11] Year: 2019	DCN + BiLSTM	Person Organization Location Object	Scheme: IOB Tag Set: 9	Newspaper, Banglapedia	Train Data: N/A Test Data: N/A	P-69.41% R-57.20% F1-62.70%
Rifat et al. [10] Year: 2019	BGRU + CNN	Person Organization Location Time	Scheme: IOB Tag Set: 8	Newspaper	Train Data: N/A Test Data: N/A	P-73.32% R-72.27% F1-72.66%
Ashrafi et al. [12] Year: 2020	BERT + BLSTM + CRF + CW	Person Location Organization Object	Scheme: BIOES IOB Tag Set: 9	Newspaper, Banglapedia	Train Data: 8,85,090 (token) Test Data: 49,286	P-65.60% R-66.78% F1-65.96%

2.4 Scope of the Problem

In our study, we used Deep Learning to build models for recognizing Named Entities in Bangla text. We choose this because, in comparison to previous work progress, deep learning and automated processes have increased our working and processing capability. However, our proposed model finds the entity (such as person, organization, location, quantity, percentage, currency) from the Bangla text document. For the Bangla language, this can solve problems like knowledge retrieval, computer translation, question answering, text summarization, and so on. For the advancement of the Internet and technology, several journals, blogs, tweets, and newspapers are published online. This massive volume of text on the internet remains unstructured. This information is easier to comprehend for humans. On the other hand, machines are unable to comprehend this unstructured data. As a result, we must make this data machine-friendly so that the machine can retrieve information quickly. However, in comparison to other languages such as English, named entity recognition in Bangla is more difficult. In English, a noun word begins with a capital letter, whereas in Bangla, this is not the case. To escape different problems and behave appropriately in light of the Bangla language's diversity, we must think like linguists. Deep learning needs a lot of good data and a lot of it to achieve reasonable efficiency and precision. We examined previous works and discovered that the volume of data contained in them was not particularly large. In comparison, the sum of data we have is adequate to complete a NER mission in Bangla.

2.5 Challenges

A deep learning model requires a well-developed dataset to achieve a high level of accuracy in the Named Entity Recognition. Bangla is a low-resource language. In this middle of low resources, Bangla named entity recognition is being worked on. In recent times, various tools have been created for processing Bangla language with intelligence. For Bangla NER, there is a limited but well-annotated publicly available dataset. That is why we tried to create a decent and well-developed dataset from scratch. Labeling such a large amount of data was extremely difficult and time consuming. In our huge dataset, we could not automatically remove all the garbage data. For this, we had to make corrections

manually. We saw some sentences repeated while we annotated the dataset. In this case, we have omitted them and added new sentences to our dataset. There was a gap between the two sentences of our dataset and when we annotated the dataset, then we accidentally gave the 'O' tag in these blank cell areas. We have used capital letters in all the annotations of our dataset. But sometimes we accidentally use small letters instead of capital letters while doing data annotation; for example, we had to tag 'O' but we accidentally tagged 'o'. We corrected this type of error through automation. Training such a large dataset was a time-consuming task. While feeding the data to the machine, we had to check after a while whether the machine was being trained properly or not. We needed to use a decent quality CPU and GPU to train the dataset. Table 2.2 highlights some of the challenges-

TABLE 2.2: CHALANGING SEQUENCES

Multiple Meaning	আমার নাম বকুল।
	বকুল ফুলের সুবাস বেশ মিষ্টি।
Sentence Structure	ঢাকা বাংলাদেশের রাজধানী।
	পাত্রটি ঢাকনা দিয়ে ঢাকা আছে।
Proverbs	আমি রাবণের চিতায় জ্বলছি।
	আমিই সেই রাবণ।
Confusing Word	লোকটির বাড়ি কুড়িগ্রামে।
	লেবুর দাম কুড়ি টাকা।
Word Inflection	গ্রামটির নাম আড়াইহাজার।
	জামাটির দাম আড়াইহাজার।
Multiple Expression	আমি ১ লাখ টাকা ডিবিবিএল ব্যাংকে জমা করেছি।
	আমি এক লাখ টাকা ডিবিবিএল ব্যাংকে জমা করেছি।
	আমি ১,০০,০০০ টাকা ডিবিবিএল ব্যাংকে জমা করেছি।

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

The aim of this analysis is to develop an efficient system for categorizing the named entity classes from the Bengali text data. However, large well annotated datasets for NER research in Bengali are scarce. For this reason, before implementation, our dataset has been processed. The classification models are created with minimal resources such as word embedding, character embedding. This research has explored various neural network-based models such as BGRU+CNN, BLSTM+CNN+Dropout, BLSTM+CNN-Dropout, BLSTM+CNN+CRF, BGRU+CNN+CRF, BLSTM, BGRU, BLSTM+CRF, BGRU+CRF. The first five models are designed using both word and character embedding. Only word embedding is used in the remaining four models. This analysis also shows the combined impact of word embedding and character embedding to improve performance. We calculate and compute the accuracy, precision, recall, and F1 score to select the highest performing model. We found that BGRU+CNN+CRF has the highest F1 score.

3.2 Workflow

Figure 3.1 illustrates a logical diagram of the whole workflow. Data collection, Data Pre-processing, Dataset preparation, Model building, Comparative analysis, Model Evaluation, and Decision Making are the seven main segments of our workflow. The portions mentioned above will be explored in greater depth later.

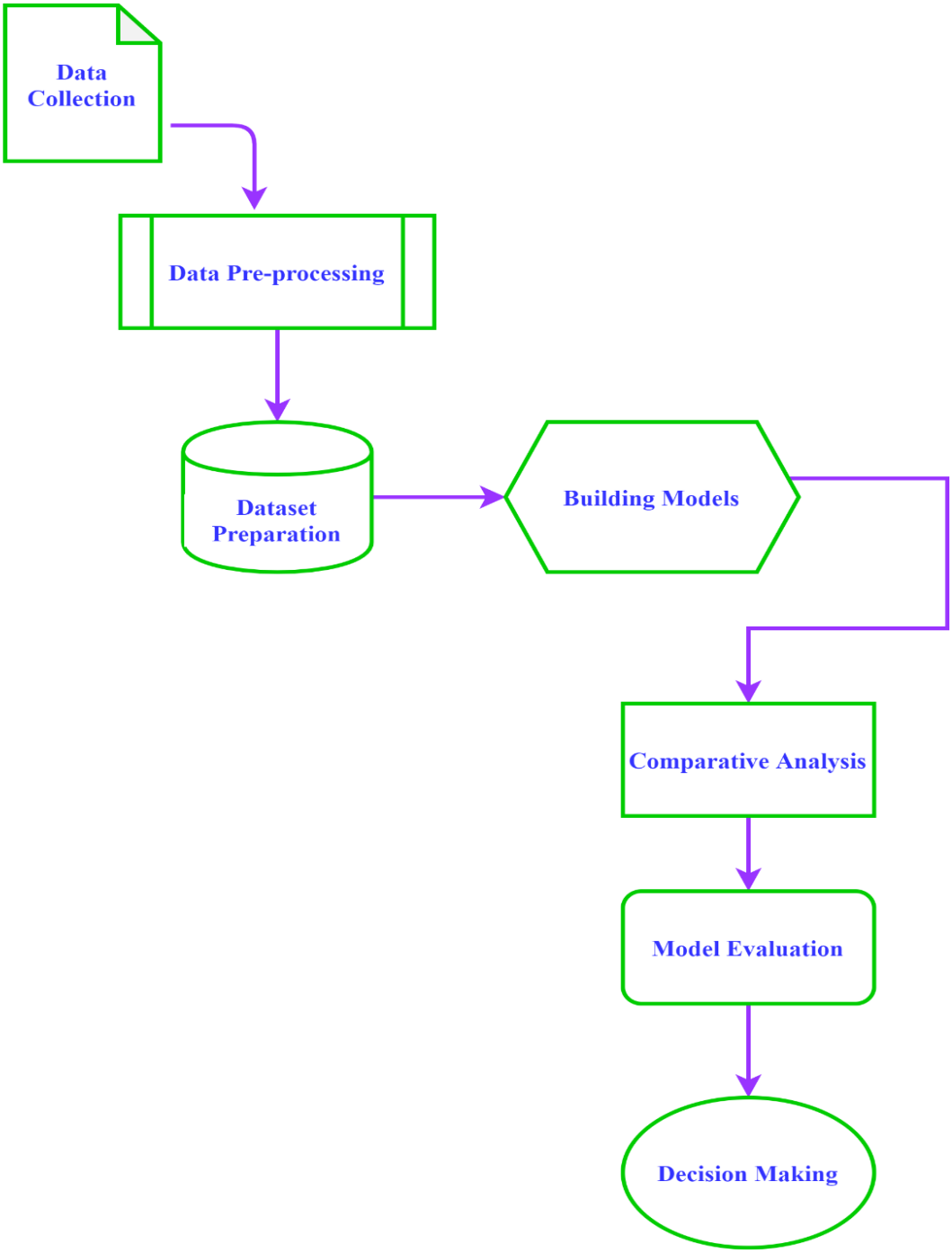


Figure 3.1: Steps regarding our research

3.3 Data Collection

Numerous studies have shown that deep learning models trained on a large amount of data achieve a high accuracy level. The amount of data required can be determined by analyzing the complexity of the task. NER is a complex sequence labeling task in which the dataset tends to skew towards the non-named entity class. Therefore, we need a large amount of training data to build a model that delivers decent performance. In this respect, online platforms are the preferred solution. Thus, the research started with collecting text documents written in Bangla from different online sources to develop a well-annotated unique dataset. Articles were scraped from the Banglapedia as well as Bangla newspaper outlets such as Prothom Alo, Bangladesh Pratidin, and Ittefaq. The sources of these websites are included in Appendix B.

Scraping is the most efficient way of collecting a large number of articles in a short time. We observed the architecture of the mentioned websites and concluded that two separate web scrapers are required. One scrapes newspaper outlets, while the other scrapes Banglapedia. We made a scraper using a widely used web scraping and crawling Python library, BeautifulSoup, which is light on memory and processor use for scraping newspaper portals. Octoparse, a versatile web scraping tool that consumes less memory and CPU power, was used to extract text from Banglapedia web pages. (See Table 3.1)

To avoid any sort of bias, we collected articles on various topics. The scraped articles cover the following areas – National news, International news, Science, Health, Sports, Editorial, Entertainment, Economy, Education, and Politics.

TABLE 3.1: ARTICLE STATISTICS

Website	Article Count
Prothom Alo	300
Bangladesh Pratidin	155
Ittefaq	300
Banglapedia	250
Total	**1005**

3.2 Data Pre-processing

The raw data has gone through the garbage elimination phase twice. Prior to tokenization, URLs and letters in any language other than Bangla have been excluded using Python's Regular Expression module. Punctuation marks were separated from words by a space. For Example,

Before separation of punctuations: অন্ন, বস্ত্র, বাসস্থান, শিক্ষা, চিকিৎসা হলো জীবনধারণের মৌলিক উপকরণ।

After separation of punctuations: অন্ন , বস্ত্র , বাসস্থান , শিক্ষা , চিকিৎসা হলো জীবনধারণের মৌলিক উপকরণ ।

During annotation, unnecessary sentences and words have stripped from the text. We got 77275 sentences as a consequence of this filtering process, with sentences varying in length from 2 to 145. With the NLTK tokenizer, we tokenized each sentence and placed each word in a new line. A new blank line has added at the end of each sentence as a sentence indicator. After tokenization কুটির শিল্প বাংলাদেশের ঐতিহ্যবাহী একটি শিল্প । this sentence turned like Table 3.2.

TABLE 3.2: SENTENCE AFTER TOKENIZATION

কুটির
শিল্প
বাংলাদেশের
ঐতিহ্যবাহী
একটি
শিল্প
।

Table 3.3 contains statistics related to dataset.

TABLE 3.3: DATASET STATISTICS

#	Frequency
Total Number of Sentence	77275
Total Number of Tokens	1005797
Unique Tokens	64958
Sentence Length	3~145

3.3 Dataset Preparation

We manually annotated all sentences following the IOB2 (Inside-Outside-Beginning) format. Our tag set consists of 13 tags – B-PER, I-PER, B-LOC, I-LOC, B-ORG, I-ORG, B-QTY, I-QTY, B-PCT, I-PCT, B-CUR, I-CUR, O. Table 3.4 contains a detailed description of the tagging style.

TABLE 3.4: ANNOTATION SCHEME

Class Name	Example	Explanation
B-PER	কলিমউদ্দিন দফাদার	Implies the starting of a person name
I-PER	কলিমউদ্দিন দফাদার	Tags the inside of a multi word person name
B-LOC	উত্তর শিয়াচর	Implies the starting of a location name
I-LOC	উত্তর শিয়াচর	Tags the inside of a multi word location name
B-ORG	ঢাকা কলেজ	Implies the starting of an organization name
I-ORG	ঢাকা কলেজ	Tags the inside of a multi word organization name
B-QTY	১০০ টন	Implies the starting of a quantity indicating phrase
I-QTY	১০০ টন	Tags the inside of a quantity indicating phrase
B-CUR	১০০ টাকা	Implies the starting of a currency indicating phrase
I-CUR	১০০ টাকা	Tags the inside of a currency indicating phrase
B-PCT	শতকরা ৫০ ভাগ	Implies the starting of a percentage indicating phrase
I-PCT	শতকরা ৫০ ভাগ	Tags the inside of a percentage indicating phrase
O	উপসংহার	Marks punctuations and anything except the above mentioned categories

Initially, the annotation procedure involved more than one human annotator. Inconsistency in tagging was noticed due to the fact that different people think from different viewpoints. A Bangla language expert re-checked and finalized all the tags to minimize this inconsistency, making necessary corrections. Figure 3.2 illustrates that 88.8% of tokens fall under the Non-named Entity class, while 12.2% belong to Named Entity classes.

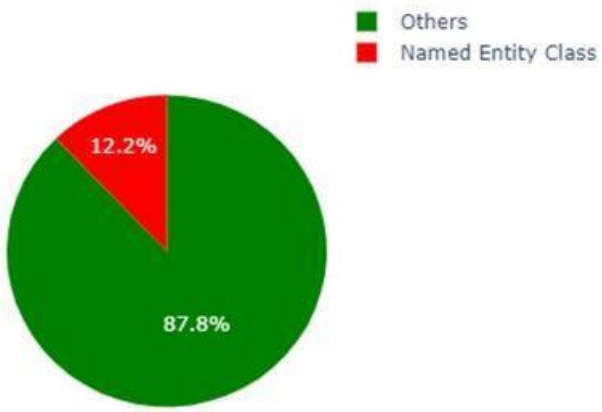


Figure 3.2: Named entity vs non-named entity ratio

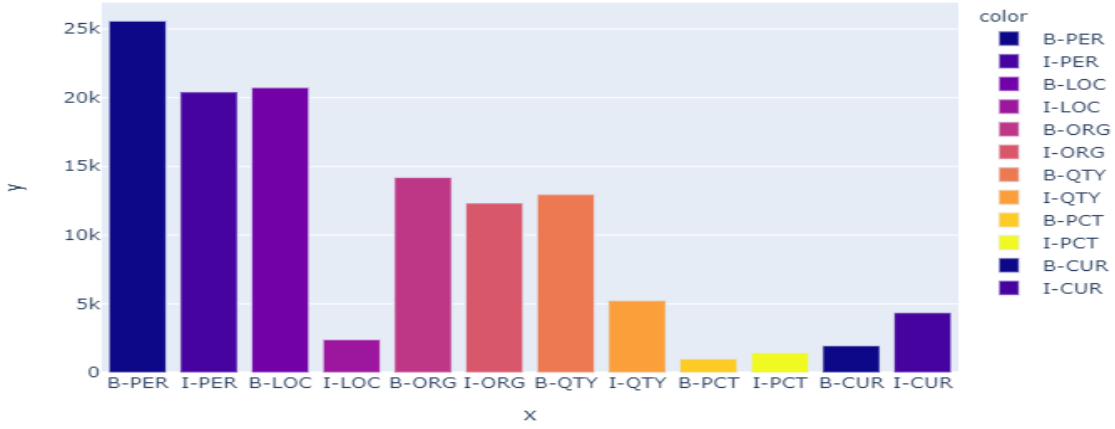


Figure 3.3: Frequency of named entity class

The frequency of each named entity class is shown in Figure 3.3. It is obvious that the beginning part of the person, location, organization, and quantity sections has a higher frequency than the inside part, while the percentage and currency sections have the

opposite. We split our dataset into the training set and testing set. The training set contained 80% of the dataset, and the testing set contained the remaining 20%. Table 3.5 includes the distribution of each class for both the training set and test set.

TABLE 3.5: CLASS DISTRIBUTION

Section	Class	Training set	Testing set
Person	B-PER	21220	4358
	I-PER	17074	3344
Location	B-LOC	16016	4723
	I-LOC	1990	406
Organization	B-ORG	11870	2328
	I-ORG	10121	2212
Quantity	B-QTY	10540	2416
	I-QTY	4130	1096
Percentage	B-PCT	728	253
	I-PCT	1075	348
Currency	B-CUR	1605	342
	I-CUR	3657	692
Others	O	704611	178642
		Total = 804637	Total = 201160

3.4 Model Overview

We developed different deep learning models based on different non-contextual word embeddings that met two criteria. One uses both word embedding and character embedding while the other only word embedding. In this section, the winning model (BGRU-CNN-CRF) with the best accuracy is going to be described in detail.

At first, sentences were taken from the training dataset and inserted into the pretrained word embedding (word2vec) to extract world level feature representations. To extract features at the character level, a time-distributed Convolutional Neural Network layer with the activation function tanh was used. Then the output from the time distributed maxpooling was fitted into a layer to flatten the neural network. Concatenation of world-level and character-level feature representations was used to send into a Bidirectional GRU network. The BGRU layers outputs were fitted into a time-distributed dense layer using the softmax activation function. Finally, we obtained the expected output after it passed through a CRF layer. This model completely avoids the dropout concept. To compile the model, 'nadam' was used as an optimizer, and crf_loss and crf_accuracy was respectively used as loss and metrics. (See Figure 3.4)

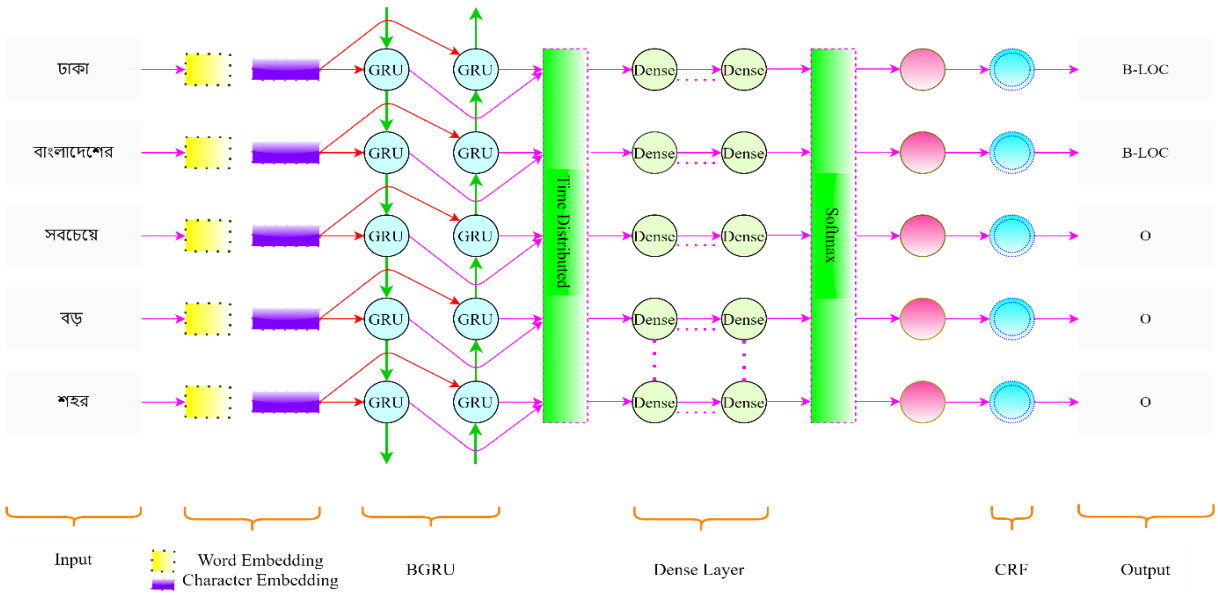


Figure 3.4: Model Architecture

Table 3.6 shows the ‘Output Shape’ and the amount of ‘Parameter’ in each layer of our proposed model which uses word2vec as non-contextualized word embedding.

TABLE 3.6: SUMMARY OF THE MODEL

Layer	Output Shape	Parameter
Words_Input (Input Layer)	(None, None)	0
Time_Distributed_1	(None, None, 1, 30)	0
Embedding_1	(None, None, 300)	15555000
Time_Distributed_2	(None, None, 1, 30)	0
Char_Input (Input Layer)	(None, None, 89)	0
Character_Embedding	(None, None, 89, 30)	3090
Time_Distributed_3	(None, None, 89, 30)	2730
Concatenate	(None, None, 330)	0
BGRU	(None, None, 400)	637200
Time_Distributed_4	(None, None, 50)	20050
CRF	(None, None, 13)	858
Total Parameters: 16,218,928		

3.5 Implementation Requirements

a) Word Embedding

One of the most important aspects of NLP study is the choice of word embedding. We used pre-trained word embedding in each model we trained to find the most accurate model. According to Ritu et al. [28], the accuracy of the NER scheme differs depending on the term embedding. In recent years, three non-contextualized embeddings for Bangla have grown in popularity, namely Word2Vec, GloVe, and fastText. Appendix A contains the sources for these word embeddings.

We used BNLP's freely accessible Word2Vec which was trained with the Bengali Wikipidea Dump Dataset. The embedding dimension is 300d and the vocab size is

1171011. The word embedding model was developed by following the skipgram method. Our dataset includes a large number of rare words and skipgram technique can deliver well performance for the rare words or phrases.

A publicly available GloVe word embedding has also used for each and every model. The Glove word embedding model was trained with the Bengali Wikipidea Dump Dataset. The embedding dimension is 300 and the vocab size is 39M.

In addition, a publicly available fastText word embedding has employed for each model. The fastText word embedding model was trained with the Common Crawl and Wikipidea. The embedding dimension is 300 [29].

Our entire dataset contains 64958 unique words. Table 3.7 shows among the unique words how many number of tokens found in each word embedding. As it can be observed that matching percentage for every word embedding is below 90%. For this reason, we tuned hyper parameter by setting trainable to true that results in updating the weights of word vectors in real time according to the data. Our best performed model uses the Word2Vec.

TABLE 3.7: STATISTICS OF MATCHED WORD IN EACH WORD EMBEDDING

Word Embedding	Number of Matched Words	Percentage
Word2Vec(skipgram)	51850	79.8%
GloVe	46351	71.4%
fastText	54377	83.7%

b) Character Embedding

For the first time, Chiu et al. [15] used character embedding in English NER and achieved state-of-the-art performance. Driven by this, Rifat et al. [10] created a model in Bangla NER that used character embedding and obtained a decent F1 value on a small dataset. To compare the impact of character embedding in a large dataset, we trained some models with character embedding while others were trained without.

Initially a character set is created which includes all the unique characters present in our dataset. There are 89 different characters in this collection, including punctuation marks and special characters. Unknown tokens are also considered and integrated into a single embedding. The dimension of the character embedding is 30 and the values were chosen at random with a uniform distribution ranging from -0.5 to 0.5 [10].

c) Convolutional Neural Network (CNN)

The Convolutional Neural Network employs convolution instead of matrix multiplication. The application of CNN and its variations plays an important part in Computer Vision. Both Chiu et al. [15] and Ma et al. [30] have discovered that the performance of the sequence labeling task can be improved by extracting character level feature using CNN. In Bangla NER, Karim el al. [11] used a Densely Connected Network on a large dataset to extract character level features, whereas Rifat et al. [10] used CNN on a small dataset to extract character level features. We would not make any comparisons between these two studies because their tag sets differed.

Bangla words have a dynamic nature. The addition of a suffix or a prefix at the root of a word can change the meaning of the word depending on the context of other words in a sequence. Table 3.8 illustrates how suffixes and prefixes alter the meaning of root words.

Driven by these, in our research, we have used CNN to extract character level features to make understand our models the complex structure of Bangla words. Each word has passed through a convolution layer with a maxpooling layer to create new feature map from the character embedding. The entire procedure is depicted in Figure 3.5.

TABLE 3.8: CHANGING NATURE OF ROOT WORD WITH SUFFIX & PREFIX

Area	Root Word	Suffix	Prefix	Converted Word
Person	রাম	অঘা	-	অঘারাম
	জামাল	-	পুর	জামালপুর
Location	পূর্বাচল	-	এ	পূর্বাচলে
Organization	বিদ্যালয়	-	এর	বিদ্যালয়ের
Quantity	এক	-	শ	একশ
Percentage	১০০	-	%	১০০%
Currency	টাকা	-	র	টাকার

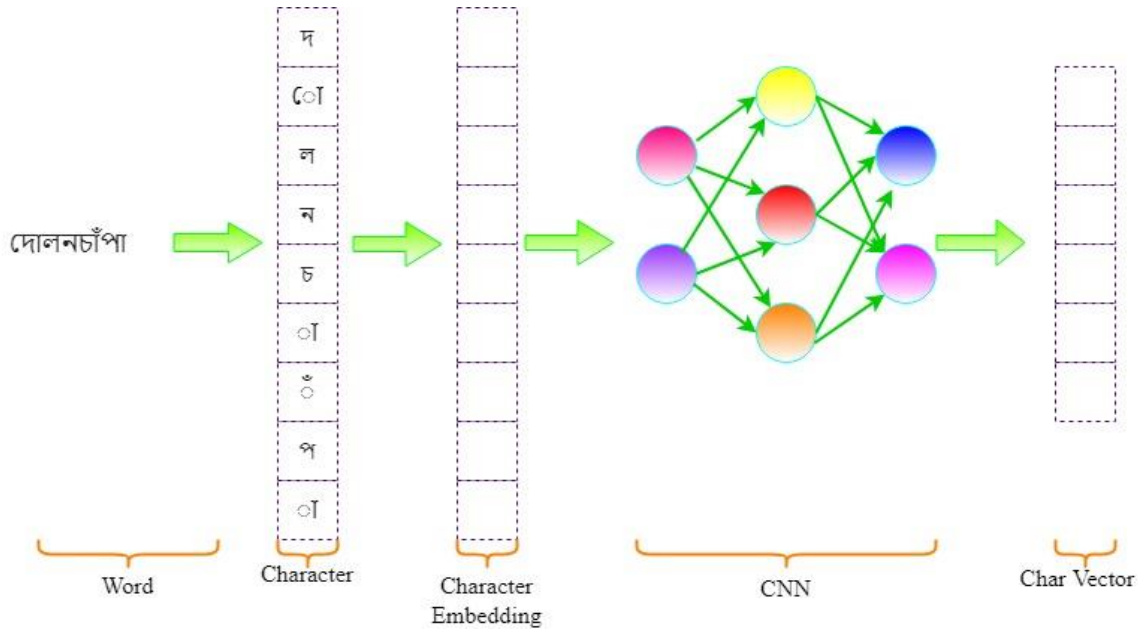


Figure 3.5: Character level feature extraction

d) Bidirectional Gated Recurrent Unit (BGRU)

Recurrent Neural Network has multifarious use in Natural Language Processing. RNN comes up with a memory facility that allows the previous input to influence future predictions. If ((রমিজ উদ্দীন একজন দার্শনিক ছিলেন), (Ramiz Uddin was a philosopher)) is a sentence and we use RNN for NER task, RNN can identify রমিজ উদ্দীন as a person entity. Because, it is a small sentence. But if ((রমিজ উদ্দীনের স্মৃতি সংরক্ষণ ও মুক্তিযুদ্ধে অসামান্য অবদান পালন করার জন্য ১৯৯৮ সালে শহীদ বীর বিক্রম রমিজ উদ্দিন ক্যান্টনমেন্ট কলেজ স্থাপন করা হয়), (Shaheed Bir Bikram Ramiz Uddin Cantonment College was established in 1998 to preserve the memory of Ramiz Uddin and to make outstanding contribution in the war of liberation)) is a sentence and we use RNN, RNN may identify রমিজ উদ্দীন as a person entity but RNN will not be able to identify শহীদ বীর বিক্রম রমিজ উদ্দিন ক্যান্টনমেন্ট কলেজ as an organization entity because of vanishing gradient problem. To solve this issue, many variants of RNN has been discovered over times.

By learning long term dependencies, Long Short Term Memory Network removes the shortcoming of RNN. LSTM has three gates where the cell state acts as memory and transfers the related information to the next part of the sequence. GRU is simpler version of LSTM having only two gates. GRU uses hidden state rather than cell state to transfer the related information to the next part of the sequence. Bidirectional GRU identifies a word in a predefined class by considering the context of the previous and next words in a sequence. GRU is faster and uses less memory than LSTM. In our study, we have used both LSTM and GRU to compare the performance.

e) Conditional Random Field (CRF)

It is important to consider contextual information in the sequence labeling task in order to make a correct prediction. In consequence, we have combined CRF with LSTM and GRU in some of our models. Our best performing model uses Linear chain CRF. If a sentence is ((রহমত বুদ্ধিমান ছেলে), (Rahmat is an intelligent boy)), CRF will take

neighbor entities information for each entity in the sequence and provide highest probability for [B-PER, O, O] as compared to other possible tag sequences such as [I-PER, O, O], [O, O, O] etc.

3.6. Instrumentation

Hardware Requirements:

- Processor: 3.0 GHz, 4.0 core CPU and multithreading enabled
- GPU: NVDIA GTX1050
- Memory: 12GB of physical RAM
- Storage: 20GB of secondary space (SSD/HDD)

Software Requirements:

- Octoparse
- Operating System: Windows 10
- Excel
- Notepad++

Required Environments:

- Anaconda
- Jupyter Notebook
- NVIDIA CUDA Toolkit v-10.2
- NVIDIA CUDNN Toolkit v-8.0

Developing Tools:

- Python 3.7
- Tensorflow
- Keras
- Plotly
- Numpy
- Matplotlib
- Pandas
- Scikit-learn

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Introduction

NER task is extremely complex because the dataset is very skewed to non-named entity classes. To evaluate a task performed on an imbalanced dataset, custom evaluation metrics are needed. Precision, Recall, F1, Confusion Matrix are the appropriate evaluation matrices for the model we developed. Additionally, the challenges introduced in Table 2.1 were tested by our best performing model. We have also compared the effects of different embedding types on model performance.

4.2 Experimental Results

Figure 4.1 depicts the categorization of models depending on the type of embedding was used for their implementation.

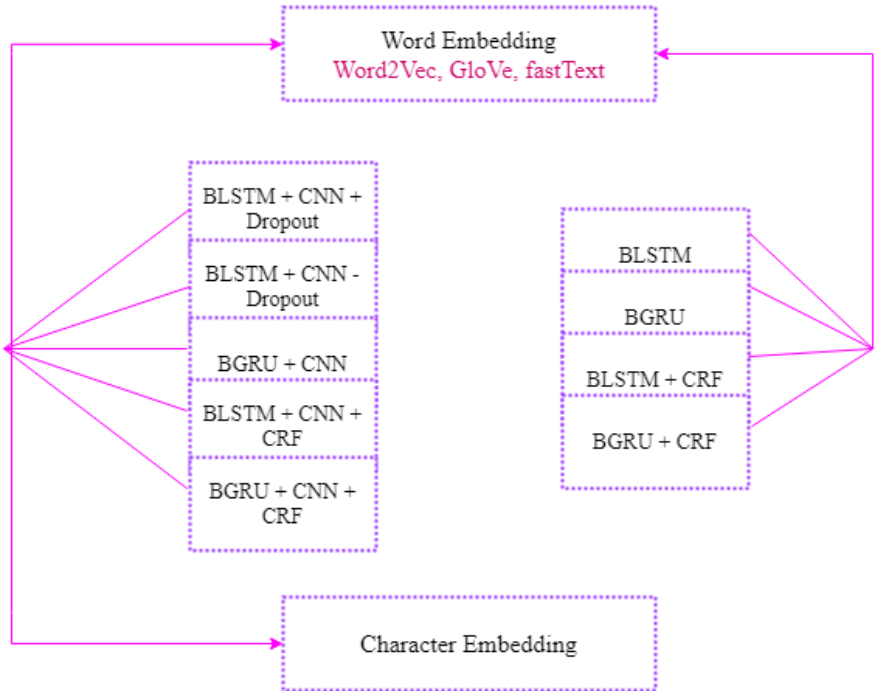


Figure 4.1: Categorization of models according to the uses of embedding

a) Precision, Recall, F1:

Precision calculates the number of true positive class predictions that actually belong to the positive class [31]. In our dataset, there are 13 classes where 87.8% tokens in the 'Other' class and 12.2% tokens in the other 12 classes. In this case, the calculation of Precision follows the following formula-

$$Micro\ Precision = \frac{\sum_{C=1}^{12} True\ Positives}{\sum_{C=1}^{12} True\ Positives\ +\ \sum_{C=1}^{12} False\ Positives}$$

This is the formula for calculating the Micro Precision where c stands for class. Micro precision is concern about label imbalance. However, following is the formula of calculating the Macro Precision for our problem-

$$Macro\ Precision = \frac{\sum_{C=1}^{12} Precision}{12}$$

Recall calculates the number of positive class predictions made out of all positive examples in the dataset [31]. The method calculating Micro and Macro Recall is as follows-

$$Micro\ Recall = \frac{\sum_{C=1}^{12} True\ Positives}{\sum_{C=1}^{12} True\ Positives\ +\ \sum_{C=1}^{12} False\ Negatives}$$

$$Macro\ Recall = \frac{\sum_{C=1}^{12} Recall}{12}$$

F1 score is the harmonic mean of Precision & Recall. Micro and Macro precision can be calculated by the following method.

$$Micro\ F1 = 2 * \frac{\text{Micro Preicision} * \text{Micro Recall}}{\text{Micro Preicision} + \text{Micro Recal}}$$

$$Macro\ F1 = 2 * \frac{\text{Macro Preicision} * \text{Macro Recall}}{\text{Macro Preicision} + \text{Macro Recal}}$$

Table 4.1 - 4.9 includes Micro and Macro Precision, Recall and F1 score of each model under different non-contextual word embeddings and character embeddings.

TABLE 4.1: PRECISION, RECALL, F1 SCORES (Word2Vec & CHARACTER EMBEDDING)

Model Name	Embedding (W: Word, C: Character)	Custom(%)			Sk-learn (Macro)(%)		
		P	**R**	**F1**	**P**	**R**	**F1**
BLSTM+CNN+ Dropout	W + C	88.07	73.46	80.10	93.46	82.22	86.88
BLSTM+CNN- Dropout	W + C	85.48	76.79	80.90	92.16	85.80	88.64
BGRU+CNN	W + C	87.41	81.05	84.11	92.35	90.19	91.15
BLSTM+CNN+CRF	W + C	84.65	74.72	79.38	91.65	81.22	85.00
BGRU+CNN+CRF	W + C	90.34	83.59	86.83	93.81	90.20	91.90

TABLE 4.2: PRECISION, RECALL, F1 SCORES (Word2Vec)

Model Name	Embedding (C: Character)	Custom(%)			Sk-learn (Macro)(%)		
		P	**R**	**F1**	**P**	**R**	**F1**
BLSTM	C	83.90	70.9	76.90	90.87	81.78	85.73
BGRU	C	83.37	75.28	79.12	90.22	86.36	88.10
BLSTM+CRF	C	88.71	76.58	82.20	92.99	84.31	88.16
BGRU+CRF	C	86.36	79.05	82.54	91.45	87.24	89.25

TABLE 4.3: MICRO F1 SCORES (Word2Vec)

Model Name	Sk-learn P, R, F1(Micro) (%)
BLSTM+CNN+ Dropout	97.36
BLSTM+CNN- Dropout	97.56
BGRU+CNN	97.94
BLSTM+CNN+CRF	97.07
BGRU+CNN+CRF	98.21
BLSTM	97.04
BGRU	97.26
BLSTM+CRF	97.58
BGRU+CRF	97.64

TABLE 4.4: PRECISION, RECALL, F1 SCORES (GloVe & CHARACTER EMBEDDING)

Model Name	Embedding (W: Word, C: Character)	Custom(%)			Sk-learn (Macro)(%)		
		P	**R**	**F1**	**P**	**R**	**F1**
BLSTM+CNN+ Dropout	W + C	88.15	72.76	79.72	94.10	83.31	87.79
BLSTM+CNN- Dropout	W + C	86.81	77.00	88.61	92.87	86.43	89.24
BGRU+CNN	W + C	86.03	79.62	82.70	92.47	88.89	90.52
BLSTM+CNN+CRF	W + C	86.66	82.16	84.35	91.75	90.24	90.92
BGRU+CNN+CRF	W + C	89.49	84.53	86.94	93.62	91.09	92.31

TABLE 4.5: PRECISION, RECALL, F1 SCORES (CHARACTER EMBEDDING)

Model Name	Embedding (C: Character)	Custom(%)			Sk-learn (Macro)(%)		
		P	**R**	**F1**	**P**	**R**	**F1**
BLSTM	C	83.00	73.00	77.60	91.51	84.73	87.76
BGRU	C	82.72	73.42	77.79	90.91	85.61	88.09
BLSTM+CRF	C	85.92	77.34	81.40	91.84	86.34	88.84
BGRU+CRF	C	84.71	75.95	80.09	90.30	86.83	88.43

TABLE 4.6: MICRO F1 SCORES (GloVe)

Model Name	Sk-learn P, R, F1(Micro) (%)
BLSTM+CNN+ Dropout	97.34
BLSTM+CNN- Dropout	97.64
BGRU+CNN	97.74
BLSTM+CNN+CRF	97.88
BGRU+CNN+CRF	98.23
BLSTM	97.14
BGRU	97.15
BLSTM+CRF	97.47
BGRU+CRF	97.29

TABLE 4.7: PRECISION, RECALL, F1 SCORES (fastText & CHARACTER EMBEDDING)

Model Name	Embedding (W: Word, C: Character)	Custom(%)			Sk-learn (Macro)(%)		
		P	**R**	**F1**	**P**	**R**	**F1**
BLSTM+CNN+ Dropout	W + C	84.32	65.03	73.43	92.87	77.38	82.39
BLSTM+CNN- Dropout	W + C	84.94	61.88	71.60	92.98	74.68	81.09
BGRU+CNN	W + C	86.03	79.62	82.70	92.47	88.89	90.52
BLSTM+CNN+CRF	W + C	89.44	76.14	82.26	93.71	83.80	87.77
BGRU+CNN+CRF	W + C	89.08	77.94	83.14	92.76	85.73	88.89

TABLE 4.8: PRECISION, RECALL, F1 SCORES (CHARACTER EMBEDDING)

Model Name	Embedding (C: Character)	Custom(%)			Sk-learn (Macro)(%)		
		P	**R**	**F1**	**P**	**R**	**F1**
BLSTM	C	76.36	51.56	61.56	88.42	68.18	74.41
BGRU	C	78.71	62.14	69.45	87.11	77.67	80.95
BLSTM+CRF	C	91.88	79.87	84.70	87.71	69.51	77.56
BGRU+CRF	C	80.36	63.14	7072	89.32	70.07	77.16

TABLE 4.9: TABLE 4.6: MICRO F1 SCORES (fastText)

Model Name	Sk-learn P, R, F1(Micro) (%)
BLSTM+CNN+ Dropout	96.48
BLSTM+CNN- Dropout	96.30
BGRU+CNN	97.74
BLSTM+CNN+CRF	97.44
BGRU+CNN+CRF	97.64
BLSTM	95.12
BGRU	95.97
BLSTM+CRF	96.83
BGRU+CRF	95.96

b) Confusion Matrix

Figure 4.2 - 4.4 illustrates the Confusion Matrix for the best performing model which gives an insight into how many tokens have been correctly and incorrectly predicted in a particular class.

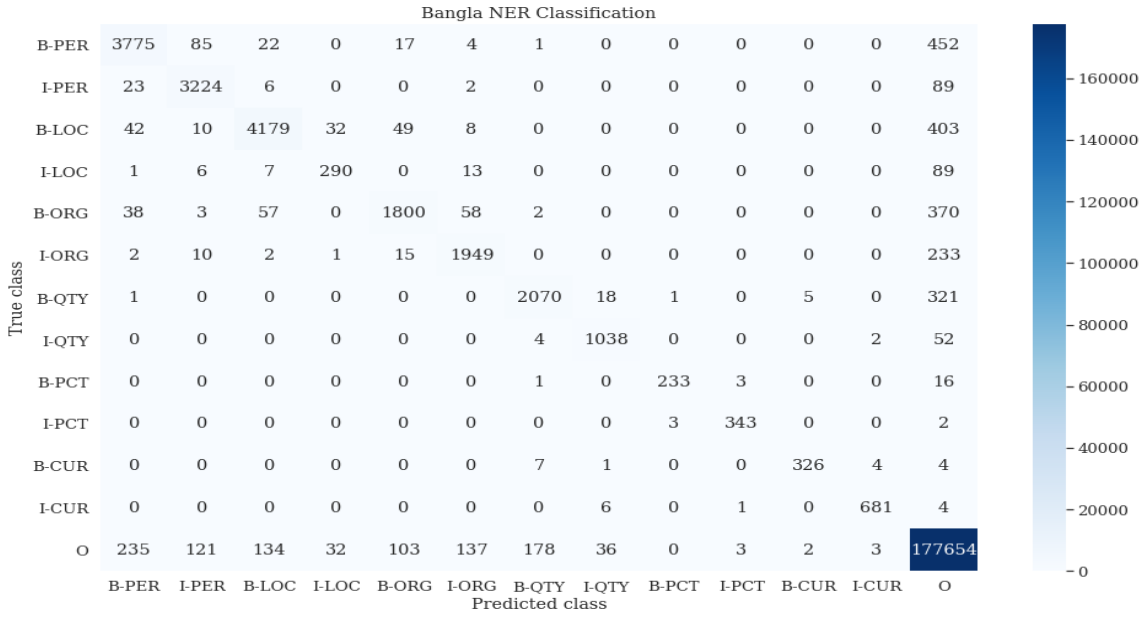


Figure 4.2: Confusion matrix of BGRU+CNN+CRF (Word2Vec)

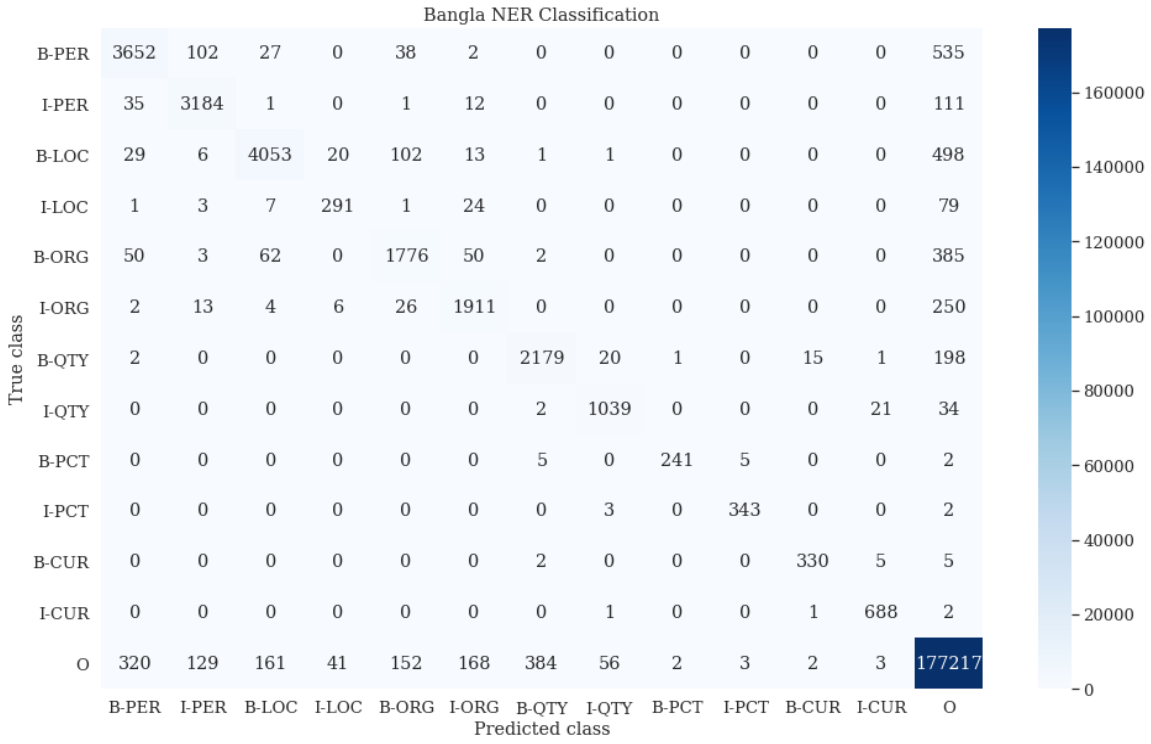


Figure 4.3: Confusion matrix of BGRU+CNN+CRF (GloVe)

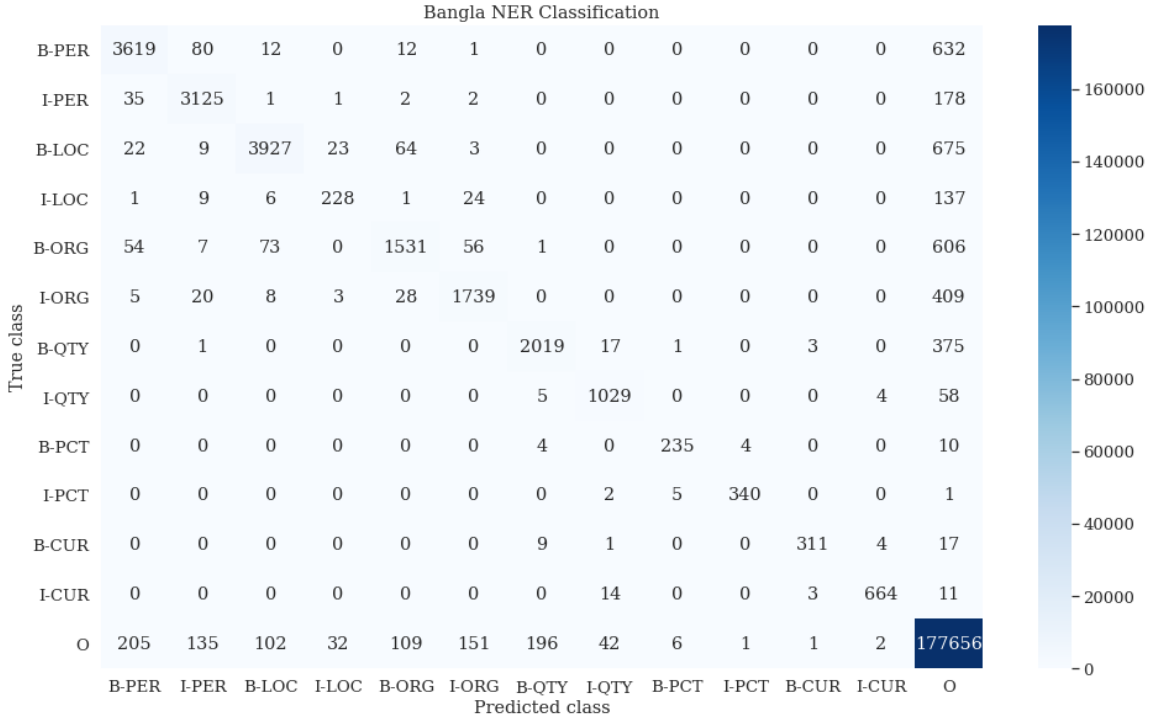


Figure 4.2: Confusion matrix of BGRU+CNN+CRF (fastText)

4.3 Comparative Analysis:

The majority of previous work used conventional machine learning techniques and hand-crafted features. Another significant point to consider is that previous experiments have focused on different entities from ours. We present a comparative analysis on evaluation with other existing systems in Table 4.10.

TABLE 4.10: RESULTS OF THE COMPARISON OF OUR WORK AND OTHERS' WORKS

Model	Method	Evaluation(%)
This work	BGRU + CNN + CRF	P- 93.81 R- 90.20 F- 91.90
Ekbal et al. [1]	HMM	P- 79.52 R- 90.30 F1- 84.5
Ekbal et al. [2]	SVM	P- 89.4 R- 94.3 F1- 91.8
Ekbal et al. [3]	CRF	P- 87.8 R- 93.8 F1- 90.7
Hasanuzzaman et al. [4]	ME	P- 82.63 R- 88.01 F1- 85.22
Ekbal et al. [5]	Voted System	P- 90.63 R- 93.98 F1- 92.28
Parvez et al. [6]	HMM	P- 85.07 R- 94.07 F1- 90

Banik et al. [8]	GRU	F1- 69
Chowdhury et al. [9]	LSTM + CRF	P- 0.65 R- 0.53 F- 0.58
Karim et al. [11]	DCN + BiLSTM	P- 69.41 R- 57.20 F1- 62.70
Rifat et al. [10]	BGRU + CNN	P- 73.32 R- 72.27 F1- 72.66
Ashrafi et al. [12]	BERT + BLSTM + CRF + CW	P- 65.60 R- 66.78 F1- 65.96

4.4 Discussion

From the Table 4.1 – 4.9, it can be observed that Dropout can change the accuracy. Models that use Dropout have improved efficiency. It is clear that BGRU-based models outperform BLSTM-based models by increasing the F1 score by 1-2%. When compared to other models that do not use CRF, CRF may have superior efficiency. The models that use both word and character level features are more powerful than the models that only use word level features. Models that have been incorporated with pretrained Wor2Vec perform best and worst with fastText.

CHAPTER 5

SUMMARY, LIMITATIONS & CONCLUSIONS, FUTURE WORK

5.1 Summary

We have successfully implemented the model for Named Entity Recognition in Bangla. The entire project summary is the followings-

Step 1: Data Collection

Step 2: Data Pre-processing

Step 3: Data Annotation

Step 4: Model Design

Step 5: Model Implementation

Step 6: Model Training

Step 7: Evaluation of Models

5.2 Limitations & Conclusions

The hardware, and GPU support specifications for training Deep Learning models are a major concern, and this is our key drawback. The atmosphere we had for training such a large dataset was insufficient for us. For training such a massive dataset, the internet must be extremely secure. Working in such a setting was very difficult. We would expect the accuracy of each of our models to improve by at least 3 to 4%, if we had a suitable setting.

Other researchers will be able to use our dataset in the future. Others will use our top-performing model for a variety of sequence labeling tasks in natural language processing. The most serious flaw in named entity identification is that the maximum class is non-

named entities. With such a bad climate, developing a NER scheme with such a heavily biased dataset was extremely challenging.

5.3 Future Work

In future, we want to increase the amount of data. We have used newspapers and Banglapedia data as data sources. We will gather data from various sources in future. In this study we have worked with six entities- person, location, organization, percentage, currency, and quantity. In future, there are proposals to work in more areas, such as date, time, object, and so on. In our study, we have used non-contextualized word embedding. In further study, we will use contextualized embedding, such as BERT.

REFERENCES

[1] A. Ekbal and S. Bandyopadhyay, "A Hidden Markov Model Based Named Entity Recognition System: Bengali and Hindi as Case Studies," in *Pattern Recognition and Machine Intelligence*, 2007.

[2] A. Ekbal and S. Bandyopadhyay, "Bengali Named Entity Recognition using Support Vector Machine," in *Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages*, 2008.

[3] A. Ekbal, R. Haque and S. Bandyopadhyay, "Named Entity Recognition in Bengali: A Conditional Random Field Approach," in *Proceedings of the Third International Joint Conference on Natural Language Processing*, 2008.

[4] M. Hasanuzzaman, A. Ekbal and S. Bandyopadhyay, "Maximum Entropy Approach for Named Entity Recognition in Bengali and Hindi," *International Journal of Recent Trends in Engineering,* vol. 1, no. 1, pp. 408-412, 2009.

[5] A. Ekbal and S. Bandyopadhyay, "Named Entity Recognition in Bengali: A Multi-Engine Approach," *Northern European Journal of Language Technology*, vol. 1, pp. 26-58, 2009.

[6] S. Parvez, "NAMED ENTITY RECOGNITION FROM BENGALI NEWSPAPER DATA," *International Journal on Natural Language Computing,* vol. 6, no. 3, p. 47–56, 2017.

[7] N. Ibtehaz and A. Satter, "A Partial String Matching Approach for Named Entity Recognition in Unstructured Bengali DataA Partial String Matching Approach for Named Entity Recognition in Unstructured Bengali Data," *International Journal of Modern Education and Computer Science,* vol. 1, pp. 36-45, 2018.

[8] N. Banik and M. H. H. Rahman, "GRU based Named Entity Recognition System for Bangla Online Newspapers," in *International Conference on Innovation in Engineering and Technology*, 2018.

[9] S. A. Chowdhury, F. Alam and N. Khan, "Towards Bangla Named Entity Recognition," in *International Conference of Computer and Information Technology*, 2018.

[10] M. J. R. Rifat, S. Abujar and S. R. H. Noori, "Bengali Named Entity Recognition: A survey with deep learning benchmark," in *International Conference on Computing, Communication and Networking Technologies*, 2019.

[11] R. Karim, M. M. Islam, S. R. Simanto, S. A. Chowdhury, K. Roy, A. A. Neon, M. S. Hasan, A. Firoze and R. M. Rahman, "A step towards information extraction: Named entity recognition in Bangla using deep learning," *Journal of Intelligent & Fuzzy Systems,* vol. 37, no. 6, pp. 7401-7413, 2019.

[12] I. ASHRAFI, M. MOHAMMAD, A. S. MAUREE, G. M. A. NIJHUM, R. KARIM, N. MOHAMMED and S. MOMEN, "Banner: A Cost-Sensitive Contextualized Model for Bangla Named Entity Recognition," *IEEE Access*, vol. 8, pp. 58206 - 58226, 2020.

[13] Y. Zhang and J. Yang, "Chinese NER Using Lattice LSTM," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018.

[14] X. Li, J. Feng, Y. Meng, Q. Han, F. Wu and J. Li, "A Unified MRC Framework for Named Entity Recognition," in *Proceedings of the 58th Annual Meeting of the Association for Computational*

Linguistics, 2020.

[15] J. P. Chiu and E. Nichols, "Named Entity Recognition with Bidirectional LSTM-CNNs," *Transactions of the Association for Computational Linguistics,* vol. 4, p. 357–370, 2016.

[16] J. Yang, Y. Zhang and F. Dong, "Neural Reranking for Named Entity Recognition," in *Proceedings of Recent Advances in Natural Language Processing*, 2017.

[17] G. Aguilar, S. Maharjan, A. P. Lopez-Monroy and T. Solorio, "A Multi-task Approach for Named Entity Recognition in Social Media Data," in *Proceedings of the 3rd Workshop on Noisy User-generated Text*, 2017.

[18] P. Cao, Y. Chen, K. Liu, J. Zhao and S. Liu, "Adversarial Transfer Learning for Chinese Named Entity Recognition," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

[19] A. Ghaddar and P. Langlais, "Robust Lexical Features for Improved Neural Network Named-Entity Recognition," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018.

[20] A. Baevski, S. Edunov, Y. Liu, L. Zettlemoyer and M. Auli, "Cloze-driven Pretraining of Self-attention Networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019.

[21] Z. Jie and W. Lu, "Dependency-Guided LSTM-CRF for Named Entity Recognition," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019.

[22] H. Chen, Z. Lin, G. Ding, J. Lou, Y. Zhang and B. Karlsson, "GRN: Gated Relation Network to Enhance Convolutional Neural Network for," in *AAAI Conference on Artificial Intelligence*, 2019.

[23] Y. Jiang, C. Hu, T. Xiao, C. Zhang and J. Zhu, "Improved Differentiable Architecture Search for Language Modeling and Named Entity Recognition," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019.

[24] J. Strakova´, M. Straka and J. Hajicˇ, "Neural Architectures for Nested NER through Linearization," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[25] A. Akbik, T. Bergmann and R. Vollgraf, "Pooled Contextualized Embeddings for Named Entity Recognition," in *Proceedings of NAACL-HLT*, 2019.

[26] H. Yan, B. Deng, X. Li and X. Qiu, "TENER: Adapting Transformer Encoder for Named Entity Recognition," *ArXiv*, vol. abs/1911.04474, 2019.

[27] X. Li, X. Sun, Y. Meng, J. Liang, F. Wu and J. Li, "Dice Loss for Data-imbalanced NLP Tasks," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[28] Z. Ritu, N. Nowshin, H. M. M. Nahid and S. Ismail, "Performance Analysis of Different Word Embedding Models on Bangla Language," in *International Conference on Bangla Speech and Language Processing*, 2018.

[29] E. Grave, P. Bojanowski, P. Gupta, A. Joulin and T. Mikolov, "Learning Word Vectors for 157 Languages," in *Proceedings of the International Conference on Language Resources and Evaluation*, 2018.

[30] X. Ma and E. Hovy, "End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF," in *Association for Computational Linguistics*, 2016.

[31] "Machine Learning Mastery," [Online]. Available: https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/. [Accessed 28-4-2021].

APPENDICES

Appendix A: Abbreviation

NER = Named Entity Recognition

CNN = Convolutional Neural Network

RNN = Recurrent Neural Network

LSTM = Long Short Term Memory

GRU = Gated Recurrent Unit

CRF = Conditional Random Field

Appendix B: Data Sources

Banglapedia = https://www.banglapedia.org/

Prothom Alo = https://www.prothomalo.com/

Bangladesh Pratidin = https://www.bd-pratidin.com/

Ittefaq = https://www.ittefaq.com.bd/

PLAGIARISM REPORT

Report

ORIGINALITY REPORT

16%	13%	9%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	4%
2	Submitted to Daffodil International University Student Paper	4%
3	arxiv.org Internet Source	1%
4	Md Jamiur Rahman Rifat, Sheikh Abujar, Sheak Rashed Haider Noori, Syed Akhter Hossain. "Bengali Named Entity Recognition: A survey with deep learning benchmark", 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019 Publication	1%
5	www.cs.uccs.edu Internet Source	1%
6	www.aclweb.org Internet Source	1%
7	Shuyi Wang, Chengzhi Zhang, Alexis Palmer. "Guest editorial", Information Discovery and	<1%

	Delivery, 2020 Publication	
8	Eamonn Newman. "Textual Entailment Recognition Using a Linguistically–Motivated Decision Tree Classifier", Lecture Notes in Computer Science, 2006 Publication	<1 %
9	www.waset.org Internet Source	<1 %
10	Submitted to Georgia Institute of Technology Main Campus Student Paper	<1 %
11	Imranul Ashrafi, Muntasir Mohammad, Arani Shawkat Mauree, Galib Md. Azraf Nijhum, Redwanul Karim, Nabeel Mohammed, Sifat Momen. "BANNER: A Cost-Sensitive Contextualized Model For Bangla Named Entity Recognition", IEEE Access, 2020 Publication	<1 %
12	Rim Koulali, Abdelouafi Meziane. "A contribution to Arabic Named Entity Recognition", 2012 Tenth International Conference on ICT and Knowledge Engineering, 2012 Publication	<1 %
13	aclweb.org Internet Source	<1 %

14	Nayan Banik, Md. Hasan Hafizur Rahman. "GRU based Named Entity Recognition System for Bangla Online Newspapers", 2018 International Conference on Innovation in Engineering and Technology (ICIET), 2018 Publication	<1 %
15	Submitted to University of Strathclyde Student Paper	<1 %
16	"Chinese Computational Linguistics", Springer Science and Business Media LLC, 2020 Publication	<1 %
17	j.mecs-press.net Internet Source	<1 %
18	aaltodoc.aalto.fi Internet Source	<1 %
19	www.mdpi.com Internet Source	<1 %
20	"Advances in Information Retrieval", Springer Science and Business Media LLC, 2019 Publication	<1 %
21	Submitted to City University of Hong Kong Student Paper	<1 %
22	"Proceedings of International Conference on Trends in Computational and Cognitive Engineering", Springer Science and Business Media LLC, 2021	<1 %

Publication

23	Asif Ekbal, Sivaji Bandyopadhyay. "Named Entity Recognition in Bengali: A Multi-Engine Approach", Northern European Journal of Language Technology, 2010 Publication	<1 %
24	Sadiq Hussain, J.J. Kuli, G.C. Hazarika. "The first step towards named entity recognition in mising language", 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), 2016 Publication	<1 %
25	Submitted to University of Stirling Student Paper	<1 %
26	Saira Jabeen, Gulraiz Khan, Humza Naveed, Zeeshan Khan, Usman Ghani Khan. "Video Retrieval System Using Parallel Multi-Class Recurrent Neural Network Based on Video Description", 2018 14th International Conference on Emerging Technologies (ICET), 2018 Publication	<1 %
27	researchspace.ukzn.ac.za Internet Source	<1 %
28	"Natural Language Processing and Chinese Computing", Springer Science and Business Media LLC, 2018 Publication	<1 %

29	docplayer.net Internet Source	<1 %
30	export.arxiv.org Internet Source	<1 %
31	ijarece.org Internet Source	<1 %
32	www.openaccess.hacettepe.edu.tr:8080 Internet Source	<1 %
33	Submitted to Cornell University Student Paper	<1 %
34	Jillur Rahman Saurav, Summit Haque, Farida Chowdhury. "End to End Parts of Speech Tagging and Named Entity Recognition in Bangla Language", 2019 International Conference on Bangla Speech and Language Processing (ICBSLP), 2019 Publication	<1 %
35	Yaoyong Li, Kalina Bontcheva, Hamish Cunningham. "Chapter 19 SVM Based Learning System for Information Extraction", Springer Science and Business Media LLC, 2005 Publication	<1 %
36	download.atlantis-press.com Internet Source	<1 %
37	en.techinco.net Internet Source	<1 %
38	ethesys.library.ttu.edu.tw Internet Source	<1 %
39	www.ajyal.sch.ae Internet Source	<1 %