**DECLARATION**

I hereby declare that I carried out the work reported in this system in the Department of Information Science under the Faculty of Information and Communication Technology, University of Technology (Yatanarpon Cyber City), under the supervision of Dr. Naw Thiri Wai Khin. I solemnly declare that to the best of my knowledge, no part of this system has been submitted here or elsewhere in a previous application for the award of a degree. All sources of knowledge used have been duly acknowledged.

..…………………………

29th August, 2024

Mabu Phong

6IST-21

**APPROVAL**

This is to certify that the system titled **“Hybrid Approach to Rawang Language Word Segmentation using Part-of-Speech Tagging”** carried out by **Mabu Phong, 6IST-21** has been read and approved for meeting part of the requirements and regulations governing the award of the degree of Bachelor of Engineering (Information Science and Technology), Department of Information Science under the Faculty of Information and Communication Technology, University of Technology (Yatanarpon Cyber City), Myanmar.

1. Dr. Nyo Nyo Yee

Professor and Head

Department of Information Science

Faculty of Information and Communication Technology ………………….

(Chairman)

1. Dr. May Mar Oo

Professor

Department of Information Science

Faculty of Information and Communication Technology ....………………

(Course Coordinator)

1. Dr. Naw Thiri Wai Khin

Associate Professor

Department of Information Science

Faculty of Information and Communication Technology …………………

(Supervisor)

1. Daw Yu Yu Win

Associate Professor

Department of Information Science

Faculty of Information and Communication Technology …….……………

(Member)

1. Daw Lei Yi Hwe

Lecturer

Department of Information Science

Faculty of Information and Communication Technology …….……………

(Member)

**ACKNOWLEDGEMENT**

First of all, I would like to express my special thanks to Dr. Aung San Lin, Rector, University of Technology (Yatanarpon Cyber City), for initiating the Bachelor program and for his kind permission to complete this thesis at the University of Technology (Yatanarpon Cyber City).

I would like to express my respectful gratitude to Dr. Thuzar Aung, Pro-Rector, University of Technology (Yatanarpon Cyber City), for giving enough consideration to her ideas and views.

I would like to express grateful thank to Dr. Nyo Nyo Yee, Professor and Head of Department of Information Science, Faculty of Information and Communication Technology, University of Technology (Yatanarpon Cyber City), for her kind guidance and encouragement in making my thesis complete successfully.

I am very grateful to Dr. May Mar Oo, Professor and Course Coordinator of Final Year (Information Science and Technology), Department of Information Science, Faculty of Information and Communication Technology, University of Technology (Yatanarpon Cyber City), for her kind guidance and patience in making my thesis complete successfully.

I am very grateful to my supervisor, Dr. Naw Thiri Wai Khin, Associate Professor, Department of Information Science, Faculty of Information and Communication Technology, University of Technology (Yatanarpon Cyber City), for her suggestion and supervision. She has been very supportive in this thesis, and also guides a lot, particularly, at the level of quality of presentation.

Very Special thanks go to Daw Khaing Zin Wint, Lecturer, English Department, University of Technology (Yatanarpon Cyber City) for her valuable supports from the language point of view in my thesis work.

I would like to thank all my teachers a lot for their mentoring, encouragement, and recommending this dissertation.

Finally, I am grateful to my parents and friends who specially offered strong moral and physical support, care and kindness, during the year of my thesis study.

**ABSTRARCT**

Languages are the cornerstone of cultural identity and heritage, preserving the history, traditions, and values of communities while connecting generations. In our digital age, converting languages into digital resources is crucial for their survival and accessibility. The Kachin ethnic group, comprising subgroups such as Jinghpaw, Rawang, Lisu, Zaiwa, Lashi**/**Lachik, and Lawngwaw**/**Maru, represents a rich tapestry of linguistic and cultural diversity in northern Myanmar. Among these, the Rawang language, spoken by a minority subgroup, faces a critical challenge due to its lack of digital representation. This is where Natural Language Processing (NLP), a branch of Artificial Intelligence (AI) that enables computers to understand and process human language, becomes essential. To address this challenge, this system tackles two fundamental NLP tasks for the Rawang language: word segmentation and Part-of-Speech (POS) tagging. Word segmentation involves breaking down continuous text into meaningful units, such as individual words or tokens, while POS tagging assigns grammatical labels (e.g., noun, verb) to each segmented word. The corpus for this system was manually collected and includes two distinct datasets: a unique word corpus with nearly 3,000 words for word segmentation, and a grammar-labeled corpus with nearly 4,000 sentences for POS tagging. The system employs the Bidirectional Maximum Matching (BDMM) algorithm for word segmentation, followed by a Conditional Random Fields (CRFs) model to assign appropriate POS tags to each word. This system produces a final output of clearly segmented Rawang words with corresponding grammatical labels, representing a significant step toward the powerful NLP tasks and preservation of the Rawang language.

**CONTENTS**

**Page**

DECLARATION i

APPROVAL ii

ACKNOWLEDGEMENTS iii

ABSTRACT iv

CONTENTS v

LIST OF FIGURES vii

LIST OF TABLES viii

LIST OF EQUATIONS ix

LIST OF ABBREVIATION AND SYMBOLS x

**CHAPTER 1 INTRODUCTION**

1.1 Objectives 3

1.2 Field Background 3

1.3 Overview of the System 5

1.4 Organization of the Thesis 6

**CHAPTER 2 THEORETICAL BACKGROUND**

2.1 The Rawang People and their Geographic Location 7

2.2 The Rawang Writing System 9

2.2.1 Rawang Alphabets 10

2.2.2 Consonants 10

2.2.3 Vowels 11

2.2.4 Clusters 11

2.2.5 Monophonic Consonants 11

2.2.6 Diphthongs 12

2.2.7 Tones 12

2.2.8 Rawang Sample Text 12

2.3 Artificial Intelligence 13

2.4 Natural Language Processing 13

2.5 Maximum Matching Algorithm 14

2.5.1 Forward Matching 15

2.5.2 Backward Matching 16

2.5.3 Bidirectional Matching 16

2.6 Graphical Modeling 17

2.6.1 Evolution of Graphical Models 18

2.7 Undirected Graphical Model 19

2.8 Markov Random Fields 20

2.9 Generative vs. Discriminative Model 21

2.9.1 Generative Model 21

2.9.2 Discriminative Model 22

2.10 CRFs in General 23

2.10.1 Feature Function 24

2.11 Feature Extraction for POS Tagging 25

**CHAPTER 3 SYSTEM DESIGN AND IMPLEMENTAION**

3.1 System Design 27

3.2 System Flow of Bidirectional Matching in Word Segmentation 28

3.3 System Flow of Training and Testing Processes in CRFs Model 29

3.4 Dataset 30

3.4.1 Unique Word Corpus 31

3.4.2 Grammar Labeled Corpus 32

3.4.2.1 Rawang POS tag sets 33

3.5 System Implementation 34

3.5.1 Text Preprocessing 34

3.5.1.1 Sentence segmentation 35

3.5.1.2 Removing special characters 35

3.5.1.3 Converting to lowercase 36

3.5.1.4 Removing Punctuations 36

3.5.2 Word Segmentation 37

3.5.3 Part-of-Speech Tagging 37

3.6 Part-of-Tagging Accuracy 38

3.6.1 Hyperparameter Tuning 38

**CHAPTER 4 CONCLUSION**

4.1 Conclusion 40

4.2 Benefits 40

4.3 Limitations 41

4.4 Further Extension 42

**REFERENCES** 44

**LIST OF FIGURES**

**Figure**  **Page**

2.1 Rawang People and their Costumes 7

2.2 Rawang Lady in Cultural Dress 8

2.3 Rawang Langue Area 9

2.4 Evolution of Graphical Models 19

2.5 Undirected Graphical Model 19

2.6 Special Case of Markov Random Fields 21

3.1 System Design 27

3.2 System Flow of Bidirectional Matching in Word Segmentation 28

3.3 System Flow of Training and Testing Processes in CRFs Model 30

3.4 Unique Word Corpus 31

3.5 Grammar Labeled Corpus 32

3.6 User Input of the System 34

3.7 Sentence Segmentation 35

3.8 Removing Special Characters 35

3.9 Converting to Lowercase 36

3.10 Removing Punctuations 36

3.11 Word Segmentation 37

3.12 Part-of-Speech Tagging 37

3.13 Test Set Confusion Matrix 39

**LISTS OF TABLES**

**Table Page**

2.1 Clusters 11

2.2 Monophonic Consonants 11

2.3 Diphthongs 12

2.4 Tones 12

3.1 Summary of Unique Words and Labeled Sentence Count by Bible Book 31

3.2 Rawang POS Tag Sets 33

3.3 Part-of-Speech Tagging Accuracy 38

3.4 Part-of-Speech Tagging Accuracy 38

**LIST OF EQUATIONS**

**Equation** **Page**

2.1 Forward Matching 15

2.2 Backward Matching 16

2.3 Bidirectional Matching 17

2.4 Factorization 20

2.5 Partition Function 20

2.6 CRFs Conditional Probability 24

2.7 Feature Function 24

**LIST OF ABBREVIATION AND SYMBOLS**

**Abbreviation/Symbol** **Description**

AI Artificial Intelligence

NLP Natural Language Processing

MMA Maximum Matching Algorithm

FM Forward Matching

FMM Forward Maximum Matching

BM Backward Matching

BMM Backward Maximum Matching

BDM Bidirectional Matching

BDMM Bidirectional Maximum Matching

MRFs Markov Random Fields

HMMs Hidden Markov Models

MEMMs Maximum Entropy Markov Models

CRFs Conditional Random Fields

POS Part-of-Speech Tagging

GUI Graphical User Interface

**CHAPTER 1**

**INTRODUCTION**

Natural Language Processing (NLP) is a rapidly evolving field at the intersection of computer science, artificial intelligence, and linguistics. Its primary objective is to enable computers to understand, interpret, and generate human language in a manner that is both meaningful and useful. By bridging the gap between human communication and computational capabilities, NLP strives to develop intelligent systems that can seamlessly interact with users in natural and intuitive ways. The increasing integration of AI-driven systems into daily human life underscores the growing importance of NLP as a core technology for human-computer interaction.

NLP is fundamentally concerned with the interaction between computers and human (natural) languages. It involves developing algorithms and models that allow machines to process and analyze large amounts of natural language data. These processes include understanding syntax, semantics, pragmatics, and discourse. The ultimate goal is to create systems that can perform complex language-based tasks, such as understanding text, generating coherent responses, and deriving meaningful insights from human communication.

The journey of NLP involves a series of foundational analysis that build upon each other to transform raw text data into structured, actionable information. Each analysis involves distinct techniques and methodologies to achieve a specific goal within the overall framework of language understanding and processing. Some NLP foundational stages are as follows:

* Word Segmentation or Tokenization: This is the initial stage in NLP, where the text is broken down into smaller units called tokens. Tokenization is crucial because it sets the groundwork for subsequent analysis.
* Part-of-Speech (POS) Tagging: Each token is assigned as a part of speech (such as noun, verb, adjective, etc.) based on its context and usage within a sentence. POS tagging is essential for understanding the syntactic structure of sentences, which in turn aids in tasks like sentiment analysis, machine translation, text summarization and information extraction.
* **Named Entity Recognition (NER)**: NER involves identifying and classifying key entities within text, such as names, dates, locations, and organizations. This stage is vital for information extraction, enabling systems to focus on the most relevant parts of a text.
* **Syntactic and Dependency Parsing**: Syntactic parsing involves analyzing the grammatical structure of a sentence, determining the relationships between words and phrases. Dependency parsing further breaks down these relationships to show which words depend on others, providing deeper insights into the sentence structure.
* **Stemming**: This stage reduces words to their base or root form by removing suffixes. For example, "running" becomes "run.". This technique is useful for normalizing text data in applications like search engines, text summarization and text mining.
* Lemmatization: This stage reduces words to their dictionary or base form (lemma) by considering the context and meaning of the word. For instance, "better" would be lemmatized to "good," and "running" would be lemmatized to "run".
* Morphology Analysis: Morphology is the study of the structure and form of words in a language, focusing on how words are formed from morphemes, the smallest units of meaning (like roots, prefixes, suffixes). It deals with the rules and patterns of word formation and modification.

For languages like Rawang, with limited existing NLP resources, establishing a solid foundation is paramount, as it sets the stage for all subsequent advancements in the field. Beginning with fundamental tasks such as word segmentation and Part-of-Speech Tagging is crucial, as these processes involve accurately dividing text into meaningful units and assigning grammatical categories to words, respectively. By mastering these essential components, we create a robust base that supports more complex NLP tasks such as machine translation, text summarization, and information retrieval, ultimately driving forward the development of comprehensive language technologies for Rawang.

Recognizing the importance of this groundwork, this system strategically prioritizes the development of these foundational analyses specifically for the Rawang language, thereby addressing its unique linguistic challenges and resource constraints. This strong foundation not only addresses immediate linguistic processing needs but also paves the way for future advancements in NLP for this under-resourced language, ensuring that Rawang can be adequately represented in the digital and technological landscapes.

**1.1 Objectives**

The objectives of this system are as follows:

* To contribute to the preservation of the Rawang language and its cultural significance by developing NLP tools that can be used for documentation, education, and research
* To raise awareness and motivate the Rawang community about the value of their language through the development of NLP technologies
* To develop accurate and efficient word segmentation algorithms for the Rawang language, considering its unique morphological and syntactic characteristics
* To create a reliable part-of-speech tagger for the Rawang language, capable of accurately identifying the grammatical categories of words in a given sentence
* To establish a robust foundation for future NLP applications in the Rawang language, such as machine translation, text summarization, and sentiment analysis.

**1.2 Field Background**

Word segmentation is a fundamental task in Natural Language Processing that involves dividing a continuous stream of text into individual meaningful words. The complexity of this task varies significantly across languages, largely depending on whether explicit word boundaries are present. In many Asian languages, such as Myanmar and Thai, Chinese, Japanese, the absence of clear delimiters and the continuous writing style without spaces make word segmentation particularly challenging, necessitating advanced techniques to accurately determine word boundaries.

Conversely, languages like English and Rawang, which use spaces or other delimiters between words and phrases, have a much simpler word segmentation process. These delimiters provide clear markers of word boundaries, enabling the straightforward identification of individual linguistic units. In Rawang, the presence of delimiters not only simplifies the segmentation task but also establishes a strong foundation for further NLP tasks, including Part-of-Speech Tagging, named entity recognition, and machine translation. This distinction underscores the crucial role of delimiters in easing the segmentation process and enhancing overall language processing efficiency.

Several algorithms have been proposed for word segmentation, including Forward Maximum Matching (FMM), Backward Maximum Matching (BMM), and Bidirectional Maximum Matching (BDMM). Forward Maximum Matching starts from the beginning of the text and iteratively tries to match the longest possible word from a dictionary. Backward Maximum Matching is similar but starts from the end of the text. Bidirectional Maximum Matching combines Forward and Backward Matching, selecting the segmentation that yields the highest overall score.

Part-of-Speech Tagging, another crucial task in NLP, involves assigning grammatical categories (e.g., noun, verb, adjective) to individual words in a sentence. Accurate POS tagging is essential for understanding the syntactic structure of a sentence and is a prerequisite for many NLP applications.

Various algorithms have been employed for POS tagging, including Hidden Markov Models (HMMs), Maximum Entropy Markov Models (MEMMs), and Conditional Random Fields (CRFs). HMMs model the sequence of words and their corresponding POS tags as a Markov process, assuming that the current POS tag depends only on the previous tag. MEMMs allow the current POS tag to depend on both the previous tag and the current word, making them more flexible. CRFs are a more general class of models that can capture arbitrary dependencies between words and their corresponding POS tags.

Given the challenges posed by the Rawang language, this system focuses on Bidirectional Maximum Matching (BDMM) for word segmentation and Conditional Random Fields for Part-of-Speech Tagging. Bidirectional Maximum Matching (BDMM) is expected to be effective in handling the ambiguous word boundaries in Rawang, while Conditional Random Fields can capture the complex syntactic dependencies and morphological patterns present in the language. By combining these powerful techniques, this system aims to develop a robust and accurate NLP system for the Rawang language, enhancing language processing efficiency and accuracy.

* 1. **Overview of the System**

This system presents a novel Natural Language Processing (NLP) system designed to enhance the processing of the Rawang language, a linguistic treasure with a rich cultural heritage. Due to the limited digital resources available for the Rawang language, a manual data collection effort was undertaken to establish two essential corpora: a unique Rawang word corpus and a grammar-labeled Rawang corpus, both derived from the Rawang Bible.

The Rawang word unique corpus serves as the foundation for word segmentation, which involves breaking down text into meaningful units of words. To achieve this, the Bidirectional Maximum Matching Algorithm (BDMA) is employed, leveraging the patterns and frequency of words within the corpus to effectively segment the text.

Simultaneously, the Rawang grammar-labeled corpus is utilized for Part-of-Speech (POS) Tagging. By training a Conditional Random Fields (CRFs) model on this annotated data, the system gains the capability to accurately identify the grammatical functions of words within Rawang sentences, such as nouns, verbs, adjectives, adverbs, conjunctions and more. The CRFs model is developed, using Python libraries, specifically scikit-learn and sklearn-crfsuite, which provide robust tools for sequence modeling and allow fine-tuning of the tagging process through hyperparameter optimization.

To ensure accessibility and ease of use, a Graphical User Interface (GUI) has been developed, using the Flutter framework. This GUI offers a user-friendly platform that enables seamless interaction with the system, allowing users to input Rawang text and visualize the results of both word segmentation and Part-of-Speech Tagging in real-time.

The GUI is integrated with a Python server powered by Flask, ensuring efficient communication and data exchange between the interface and the underlying NLP models. This integration not only facilitates smooth operation but also allows potential scalability and future enhancements of the system.

By combining advanced NLP techniques with an intuitive user interface, this system aims to provide a valuable tool for researchers, educators, and speakers of the Rawang language, contributing to the preservation and digitalization of this important cultural asset, ensuring its continued use and recognition globally.

* 1. **Organization of the Thesis**

In this system, there are four chapters.

Chapter 1 describes introduction, objectives of the thesis, field background and the overview of the system.

Chapter 2 discusses the theoretical background in details. It describes the Rawang people and their geographic location, Rawang writing system, Artificial Intelligence, Natural Language Processing, Maximum Matching Algorithms for word segmentation and Conditional Random Fields for POS tagging.

Chapter 3 provides the system design, system flows, the design and implementation of the system and user interface design of the system.

Chapter 4 presents the conclusion, benefits, limitations and further extension.

**CHAPTER 2**

**THEORETICAL BACKGROUND**

This chapter discusses the Rawang people and their geographic location, their writing system, and the detailed theoretical background applied in the system.

**2.1 The Rawang People and their Geographic Location**

The term "Rawang" is a general term which refers to a body of people who speak several (about 70) dialects (probably some could be closely related languages). Formerly, the Rawangs were referred to as **Nung**, **Kanung**, **Hkenung**, or **Ganung** by other tribes. The problem of nomenclature was started by the American missionary Robert H. Morse in his ‘Hierarchical Levels of Rawang Phonology’. He finalized at the name ‘Ganung-Rawang’ for the people and ‘Rawang’ for the languages. However, the term ‘Rawang’ is understood by the Rawangs as referring both to them and their languages, and it was officially adopted [7].



**Figure 2.1 Rawang People and their Costumes**

Robert and Betty Morse divided the Rawang people into five branches by general names which tend to differentiate the variations of culture and social structure: **Ganung**, **Rawang**, **Longmi**, **Nung** and **Tangsar**. Stephen A. Morse also created the

five branch distinction but with some changes in the branch names: **Daru** (Ganung or Ganøng), **Matwang** (Rawang), **Lungmi**, **Anung** (Nung) and **Tangsar**. This is only a general grouping and each major group comprises several subgroups. For instance, the Daru-Jerwang group comprises smaller subgroups such as Maláng, Zewàng (Jerwàng), Tashø, Dazøwàng, Taláwàng, Taluq, Akøpáy, Anàmpáy, Tarùng, etc.



**Figure 2.2 Rawang Lady in Cultural Dress**

The Rawangs are gentle, peace-loving, and law-abiding people. Originally animists, nearly all Rawang in Myanmar today are Christians. They are traditionally swidden farmers. Many still use the slash and burn form of agriculture. The Rawangs in the low valleys cultivate on irrigated farms, which are relatively small. They grow rice as their main crop; citrus fruit, vegetables, and other crops are also grown [7].

The Rawang people predominantly reside in the far north of Kachin State, Myanmar (Burma), particularly along the Mae Hka and Maeli Hka river valleys. Their territory extends eastward into the Salween valley in Yunnan Province, China, where they are known as the Dulong people, considered a subset of the Nu people, and westward into Arunachal Pradesh, India. In Kachin State, Rawang-populated townships include Putao, Machangbaw, Khonglangphu, and Sumprabum, and many Rawang also live in and around Myitkyina, as well as in other parts of Myanmar and across the globe. Globally, the Rawang population is estimated to be around 100,000.

The Rawang's geographical isolation has been shaped by great snow-covered mountains, mountainous subtropical jungles, and dense rainfall, which have kept them in almost total seclusion not only from other tribes but also from their relatives in adjoining valleys. They are bounded by various neighboring groups: the Lisu and Naxi to the east, the Maru and Lashi to the southeast, the Jingpaw to the south, the Khamti Shan to the southwest, the Mishmi (known as Manloq by the Rawangs) to the west, and the Tibetans to the north. The main area where the Rawang language is spoken is depicted below.



Rawang Language Area

**Figure 2.3 Rawang Language Area**

**2.2 The Rawang Writing System**

According to the ancient oral history of Rawang’s literature: The people of the Rawang had literature that was written on the animal skin but was lost because it was eaten by dog. Therefore, the people of Rawang had been living without literature for a long time. But now Rawangs have literature that was made the first step with Romanic alphabets by Dr. Robert H. Morse (Missionary in 1951). This primer for Rawangs is the first step towards literacy. As members of the great Union of Burma, with the aim of becoming literate in Burmese and Kachin, it is hoped the Rawangs will find that becoming acquainted with the written language in native tongue will make much easier the learning of the State and National languages. Thus, this primer is dedicated to the purpose of the Rawangs’ learning the principles of Christianity and social progress, and also to their becoming literate in State and National languages, that they may take their place as able and loyal citizens of the Union of Burma [2]. The Rawang alphabets, consonants, vowels symbols with their phonetic representation given in [ ] brackets are presented below.

2.2.1 Rawang Alphabets

The Rawang language has 26 alphabets as shown below which form 19 consonants and 7 vowels [2].

A [a] B [p/b] C/CH [tʃh] D [t/d] E [ɛ] F [f] G [k/g]

H [h] I [i] J [tʃ] K [kh] L [l] M [m] N [n]

O [ɔ] P [ph] R [r] S [s] T [t/d] U [u] V [ə]

W [w] X/SH [ʃ] Y [j] Z [z] Ø [Ɯ]

2.2.2 Consonants

There are 19 consonants as shown below.

B [p/b] C [tʃh] D [t/d] F [f] G [k/g]

H [h] J [tʃ] K [kh] L [l] M [m]

N [n] P [ph] R [r] S [s] T [t/d]

W [w] X [ʃ] Y [j] Z [z]

The Rawang alphabet includes unique consonants that represent specific phonetic sounds. Notably, **SH** and **X** represent the same sound, as do **CH** and **C**, with Rawang speakers choosing between letters based on individual or regional preferences.

2.2.3 Vowels

There are 7 vowels, all of which are monophthongs, as demonstrated below.

I [i] E [ɛ] A [a]

V [ə] U [u] Ø [Ɯ]

O [ɔ]

2.2.4 Clusters

Clusters in Rawang is combined two combined and pronounced different sound at the same time.

**Table 2.1 Clusters**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| GJ | GD | GL | GM | GR | GW | GY |  |
| KH | KJ | KC | KD | KL | KR | KW | KY |
| HL | HR | HW | HY |  |  |  |  |
| JG | JK | JW | JY |  |  |  |  |
| CH | CL | CW | CY |  |  |  |  |
| DG | DK | DH | DL | DR | DW | DY |  |
| TK | TH | TL | TR | TW | TY |  |  |
| SG | SK | SD | SL | SP | SR | SW | SY |
| NG | NK | NJ | ND | NL | NR | NW | NY |
| BH | BD | BL | BR | BW | BY |  |  |
| PG | PH | PJ | PC | PD | PT | PL | PM |
| PR | PK | PY | PW | LK | LR | LW | LY |
| RD | RT | RS | RN | RM | RL | RW | RY |
| ML | MR | MW | MY |  |  |  |  |

2.2.5 Monophonic Consonants

Monophonic Consonants in Rawang is combined two consonants and pronounced together.

**Table 2.2 Monophonic Consonants**

|  |  |  |
| --- | --- | --- |
| NG | NY | TS |

2.2.6 Diphthongs

A diphthong in Rawang is two vowels combined and pronounced together.

**Table 2.3 Diphthongs**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| II | IE | IA | IV | IØ | IU | IO |
| EI | EE | EA | EV | EØ | EU | EO |
| AI=ay | AE | AA | AV | AØ | AU | AO |
| VI=vy | VE | VA | VV=a | VØ | VU | VO |
| ØI=øy | ØE | ØA | ØV | ØØ | ØU | ØO |
| UI=uy | UE | UA | UV | UØ | UU | UO |
| OI=oy | OE | OA | OV | OØ | OU | OO=ow |

2.2.7 Tones

The Rawang sound system includes four tonal distinctions, represented as follows using the vowel '**a**' as a reference: the high falling tone is indicated by '**á**,' the mid tone by '**ā**' (often left unmarked), and the low falling tone by '**à**.' Syllables ending in a stop consonant '-**q**' carry a high tone [1].

**Table 2.4 Tones**

|  |  |  |  |
| --- | --- | --- | --- |
| **-** | \ | / | Q |
| Ī | Ì | Í | IQ |
| Ē | È | É | EQ |
| Ā | À | Á | AQ |
| V̄ | V̀ | V́ | VQ |
| Ø̄ | Ø̀ | Ǿ | ØQ |
| Ū | Ù | Ú | UQ |
| Ō | Ò | Ó | OQ |

2.2.8 Rawang Sample Text

In the Rawang language, spaces are explicitly used to separate individual words or phrases. Sample texts in Rawang are provided below.

**Pàmv̀ràé!** [ How are you! ]

**Nà bø̀ng kadø ètǿshìe?** [ What is your name? ]

**Ngà bø̀ng nø vpong ǿngàe.** [ My name is Ah Pong. ]

**2.3 Artificial Intelligence**

Artificial Intelligence (AI) has emerged as a transformative force in various fields, revolutionizing the way people interact with technology and solve complex problems. AI systems are designed to mimic human intelligence, enabling them to perform tasks that traditionally require human cognitive abilities, such as learning, reasoning, problem-solving, and perception.

At the core of AI lies the concept of machine learning, a subset of AI that focuses on developing algorithms and models that allow computers to learn from data and improve their performance on a specific task without being explicitly programmed. Machine learning encompasses various techniques, including supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning involves training models on labeled data, where each data point is associated with a correct output. This enables the model to learn patterns and make predictions on new, unseen data. Unsupervised learning, on the other hand, deals with unlabeled data, allowing the model to discover hidden structures and patterns within the data. Reinforcement learning focuses on training agents to make decisions in an environment to maximize rewards, learning through trial and error.

AI has found widespread applications in numerous domains, including Natural Language Processing, computer vision, robotics, healthcare, finance, and autonomous vehicles. In the context of this system, AI plays a crucial role in developing the NLP system for the Rawang language. The machine learning techniques employed in this system enable the models to learn from the provided data, capturing the nuances and complexities of the Rawang language [10].

**2.4 Natural Language Processing**

Natural Language Processing, or NLP, is a subfield of artificial intelligence that teaches computers to understand human language. It’s a blend of technology, smart algorithms, and the study of how to communicate. Initially, NLP tried to make sense of language by using strict rules, but language is too complex and ever-changing for such a rigid approach.

The introduction of machine learning was a turning point for NLP. Computers began to learn from large amounts of text, identifying patterns and gaining insights. This learning process didn’t rely on fixed rules but on algorithms that improved with more data. As a result, computers became more adept at processing language, leading to powerful tools for translation and voice recognition [4].

Deep learning has further advanced NLP, allowing computers to analyze language with greater depth. This has led to significant improvements in translation, understanding people’s feelings, and answering questions. These advancements have pushed the boundaries of what machines can achieve with human language.

NLP is now poised to explore new frontiers. It’s looking to integrate different types of data, enhance its learning capabilities, and ensure ethical use. The goal is not just to replicate human language abilities but to enhance them, paving the way for seamless interaction between people and machines.

As NLP progresses, it promises to change how people interact with tech. In a space where computers understand both the words and emotions, the possibilities are vast. They could aid language learning, provide instant information, and support those with speech issues. The future of NLP aims for a seamless blend of human intuition and machine intelligence, envisioning a world where talking to a computer feels as natural as chatting with a friend.

**2.5 Maximum Matching Algorithm**

Word segmentation is a fundamental task in Natural Language Processing (NLP), particularly for languages that do not use explicit word boundaries, such as Chinese, Thai, and Myanmar. The Maximum Matching (MM) algorithm, also known as the Longest Matching algorithm, is a heuristic method used to tackle this problem. It is based on the principle of maximizing the length of the matched words from a given lexicon during the segmentation process.

The MM algorithm operates under the assumption that the longer a word is, the more likely it is to be a valid token in the language. This assumption guides the algorithm to prefer longer sequences of characters as potential words when segmenting a sentence [3].

In the realm of Maximum Matching, forward and Backward Matching are commonly employed strategies, each with its own merits. Forward Matching starts at the beginning of the text and progresses forward, while Backward Matching takes the opposite approach, initiating from the end and moving backwards. These methods are efficient for many segmentation tasks; however, they are not without limitations. There are instances where relying solely on one direction may lead to suboptimal segmentation, particularly in complex linguistic scenarios where ambiguities abound. In such cases, Bidirectional Matching comes to the fore as a more sophisticated strategy. It combines the strengths of both forward and Backward Matching, evaluating the text from both ends to produce a more accurate and efficient segmentation.

2.5.1 Forward Matching

Forward Matching (FM) is a straightforward and intuitive approach to word segmentation. It begins at the start of a sentence and scans towards the end, looking for the longest word in the lexicon that matches the text at each step. This method is often favored for its simplicity and speed, making it a popular choice for initial attempts at segmentation. The formula can be expressed as:

(2.1)

where:

* is the input string to be segmented.
* is the dictionary of known words.
* denotes the substring of from the first character to the -th character.
* ∣∣ is the length of the word .

The objective of Forward Matching (FM) is to maximize the sum of the lengths of the selected words ​. This maximization strategy ensures that the segmented words cover as much of the original sentence as possible while maintaining linguistic coherence and minimizing the number of segments, which helps to avoid fragmenting the text into unrecognizable or less meaningful units. By prioritizing longer words, Forward Matching aims to produce a segmentation that closely aligns with the natural boundaries of words in the language, thereby enhancing the quality of the segmentation process.

The process of Forward Matching begins by selecting the longest word from the lexicon that matches the initial portion of the sentence. Once the first match is identified, it moves sequentially through the text, scanning the remaining portion of the sentence and iteratively selecting the longest word that fits the next segment. This process continues until the entire sentence is segmented, with each step aiming to maximize the length of the match. By iteratively adding the longest matching words, the approach ensures that the segmented output forms a contiguous, coherent, and meaningful representation of the original sentence. This method relies heavily on the structure and contents of the lexicon, as the availability of longer words directly influences the segmentation quality and accuracy [5].

2.5.2 Backward Matching

Backward Matching (FM) is another effective approach to word segmentation, often used as an alternative to Forward Matching. Instead of beginning at the start of a sentence, Backward Matching starts at the end and scans towards the beginning, looking for the longest word in the lexicon that matches the text at each step. This method is also valued for its simplicity and efficiency, making it a viable option for initial segmentation tasks [3]. The formula can be expressed as:

(2.2)

where:

* S is the input string to be segmented.
* D is the dictionary of known words.
* denotes the substring of from the -th character to the last character of .
* is the length of the word .

The goal of Backward Matching is similar to that of Forward Matching: to maximize the sum of the lengths of the selected words 𝑤𝑖​. This maximization ensures that the segmented words cover as much of the original sentence as possible while maintaining linguistic coherence [5].

The process of Backward Matching begins by selecting the longest word from the lexicon that matches the final portion of the sentence. It then continues scanning the remaining text, iteratively adding the longest matching words in reverse order until the entire sentence is segmented. This iterative approach ensures that the segmented words form a contiguous and meaningful representation of the original sentence.

2.5.3 Bidirectional Matching

In the intricate landscape of language, where structures can be as complex as a labyrinth, the efficiency of unidirectional approaches like forward and Backward Matching may falter. These methods, while robust in their own right, can stumble when faced with the multifaceted nature of linguistic constructs. It is in these scenarios that Bidirectional Matching shines, offering a more nuanced and comprehensive strategy for word segmentation.

Bidirectional Matching (BDM) is a sophisticated algorithm that merges the insights gained from both forward and Backward Matching. By initiating the segmentation process from both ends of the sentence simultaneously, it capitalizes on the strengths of each method, thereby enhancing the accuracy of the segmentation [3].

The Bidirectional Matching algorithm can be conceptualized as follows:



(2.3)w

where:

* represents the segmented words of the sentence .
* represents the segmented words using Forward Matching.
* represents the segmented words using Backward Matching.
* and are the number of segments produced by forward and Backward Matching, respectively.

**2.6 Graphical Modeling**

Graphical modeling is a powerful framework used in statistical analysis and machine learning to represent complex relationships between variables through graphs. In graphical models, nodes represent random variables, and edges capture the probabilistic dependencies between these variables.

This combination of graph theory and probability theory allows for an intuitive visualization and analysis of intricate systems with interconnected variables. Key types of graphical models include Bayesian Networks, which use directed acyclic graphs to model causal relationships, and Markov Random Fields, which employ undirected graphs to describe the associations between variables without assuming a causal direction. One of the primary strengths of graphical models is their ability to efficiently represent and compute joint probability distributions, especially in high-dimensional spaces. By exploiting conditional independence properties, graphical models decompose a complex joint distribution into a product of simpler, local distributions. This decomposition facilitates more efficient computation of marginal and conditional probabilities, which are essential in various applications such as prediction, data imputation, and probabilistic inference [4].

Additionally, graphical models allow for modularity in modeling; individual components of a system can be constructed independently and then combined, making them highly adaptable for diverse applications, including Natural Language Processing, computer vision, and network analysis.

2.6.1 Evolution of Graphical Models

The development of graphical models in Natural Language Processing (NLP) showcases a shift from simple generative models to more sophisticated conditional models that better capture the complexities of language data. This evolution begins with basic models like Naive Bayes, which rely on the assumption of feature independence given the class label. While effective for straightforward tasks, Naive Bayes lacks the ability to model dependencies between features, limiting its applicability in more complex NLP scenarios.

Hidden Markov Models (HMMs) enhance this framework by incorporating sequential dependencies, making them suitable for tasks that involve temporal data, such as Part-of-Speech Tagging and speech recognition. Generative directed models further extend the capabilities of graphical models by allowing for more complex interactions and dependencies among variables.

As the need for more nuanced modeling arose, the focus shifted to conditional models. Logistic Regression, for example, offers an improvement by modeling conditional dependencies without the restrictive independence assumptions of generative models, providing more flexibility in representing the relationships among features. Conditional Random Fields (CRFs) build on this foundation by introducing the ability to model sequences, capturing the interdependencies between sequential elements while conditioning on observed data, which is crucial for structured prediction tasks like named entity recognition and word segmentation.

General CRFs represent the culmination of this evolution, extending the flexibility of linear-chain CRFs to arbitrary graph structures. This allows for the modeling of complex interactions between variables beyond simple sequences, making them applicable to a broader range of tasks such as image segmentation and complex relational data analysis. By focusing on the conditional distribution of hidden variables given observed data, CRFs offer a powerful framework for accurately capturing dependencies and relationships within complex datasets [4].

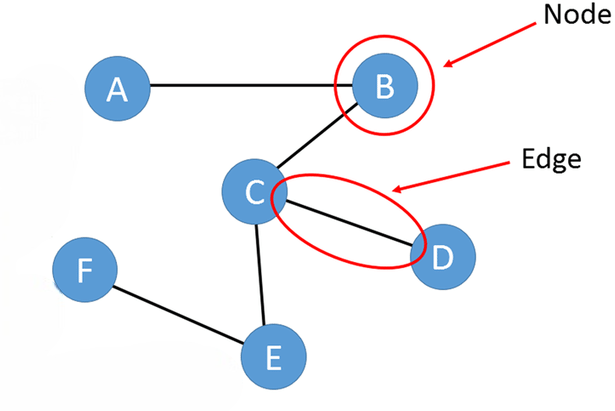


**Figure 2.4 Evolution of Graphical Models**

**2.7 Undirected Graphical Model**

Undirected graphical models are a powerful tool for representing and reasoning about complex systems with uncertainty. They are particularly well-suited for modeling pairwise relationships and global dependencies between variables.

Undirected graphical models are represented by a graph G = (V, E), where V is a set of nodes representing random variables and E is a set of edges representing pairwise relationships between the variables. The edges in an undirected graphical model are undirected, indicating that there is no inherent directionality in the relationships between the variables [9].



**Figure 2.5 Undirected Graphical Model**

A fundamental property of undirected graphical models is the concept of conditional independence, which simplifies the joint probability distribution of the variables. In undirected graphical models, the joint probability distribution of a set of random variables can be factorized into a product of potential functions (or factors) over the cliques of the graph. This factorization is expressed as:

(2.4)

where:

* denotes the set of cliques in the graph.
* is the potential function defined over the clique , representing the compatibility or strength of interaction between the variables in .

(2.5)w

* is the normalization constant, also known as the partition function, defined as:

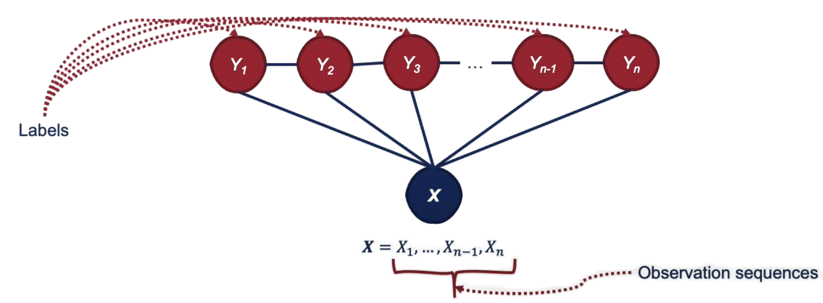
This partition function ensures that the probabilities sum to 1. The factorization into clique potentials reduces computational complexity and enhances the efficiency of probabilistic inference, making undirected graphical models an essential tool in fields that require robust handling of uncertainty and complex variable interactions.

**2.8 Markov Random Fields**

Markov Random Fields (MRFs) are a class of undirected graphical models that are particularly suited for modeling dependencies between random variables arranged in a spatial or sequential manner. MRFs are used to model complex systems by leveraging the concept of locality, where the state of a variable directly depends only on its neighbors. This local dependency structure allows MRFs to capture the essence of the problem while maintaining computational tractability.

Figure 2.6 depicts a special case of MRFs applied to sequence labeling tasks, a common scenario in Natural Language Processing and other sequential data applications. In this example, observed sequences are connected to hidden states . Each hidden variable Yi​ corresponds to an observation and interacts directly with the observation sequence as well as its neighboring hidden states. This reflects the Markov property where each label depends not only on the observed data but also on the adjacent labels, capturing the dependencies within a sequence effectively.

The undirected nature of MRFs allows the modeling of bidirectional relationships between hidden states and observations . This structure is particularly advantageous when the dependencies are not naturally directed, such as in the case of language or spatial data where influences can flow in multiple directions. By modeling these interactions through MRFs, it becomes feasible to perform tasks like labeling, segmentation, and recognition, with the graphical model facilitating the incorporation of both local and contextual information [4].

****

**Figure 2.6 Special Case of Markov Random Fields**

**2.9 Generative vs. Discriminative Model**

In the field of machine learning, models can broadly be categorized into discriminative and generative models based on how they model the data. Understanding the differences between these two types of models is crucial for selecting the appropriate approach for specific tasks, particularly in applications such as classification, sequence prediction, and structured prediction in Natural Language Processing (NLP).

2.9.1 Generative Model

Generative models focus on modeling the joint probability distribution , where represents the observed data (e.g., input features), and represents the labels or target variables. The primary goal of generative models is to understand how the data is generated by modeling the distribution of both the input features and the output labels. By learning , generative models can generate new instances of data, estimate the likelihood of observations, and perform classification by using Bayes’ theorem to compute the posterior distribution Common examples of generative models include:

* **Naive Bayes**: Naive Bayes models assume conditional independence between features given the class label, thereby making them both simple and computationally efficient.
* **Hidden Markov Models (HMMs)**: HMMs are widely used for sequence prediction tasks, modeling the joint distribution of sequences of observations and hidden states.
* **Gaussian Mixture Models (GMMs)**: GMMs are used for clustering and density estimation by modeling data as a mixture of multiple Gaussian distributions.

Generative models excel when the understanding of the data generation process is necessary or when the goal is to generate new data. However, they often make simplifying assumptions about the data distribution, which can lead to inaccuracies if these assumptions do not hold in practice [4].

2.9.2 Discriminative Model

Discriminative models, on the other hand, focus directly on modeling the conditional probability distribution , which is the probability of the labels given the observed data. Instead of trying to model how the data is generated, discriminative models are concerned solely with the decision boundary that separates different classes or labels, optimizing directly for the task at hand, such as classification or sequence labeling.Examples of discriminative models include:

* **Logistic Regression**: A linear model used for binary classification, modeling ) using a sigmoid function.
* **Support Vector Machines (SVMs)**: SVMS find the hyperplane that best separates the classes in the feature space.
* **Conditional Random Fields (CRFs)**: CRFs extend logistic regression to structured outputs, such as sequences, by modeling the dependencies between output variables and incorporating contextual information in a way that allows for more nuanced and accurate predictions.

Discriminative models are generally preferred when the primary goal is to achieve high accuracy on a specific predictive task, as they directly model the decision boundaries and do not rely on assumptions about the distribution of the inputs. This often results in better performance, especially in complex and high-dimensional spaces where generative assumptions may not hold.

**2.10 CRFs in General**

Conditional Random Fields (CRFs) are a type of discriminative undirected probabilistic graphical model used extensively in structured prediction tasks, particularly in Natural Language Processing (NLP) applications such as Part-of-Speech Tagging, named entity recognition, and syntactic parsing. CRFs provide a robust framework for modeling the conditional probability of a set of output variables given a set of observed input variables, effectively capturing the dependencies between input features and output labels while allowing the incorporation of arbitrary, overlapping, and correlated features.

CRFs are widely valued in NLP because of their ability to integrate various features of the input data, such as word context, part-of-speech tags, and morphological information, without assuming independence among these features. This flexibility allows CRFs to effectively leverage rich, overlapping, and interdependent features that other models might struggle to incorporate, thereby enhancing the model's predictive performance and robustness in structured prediction tasks.

Unlike generative models such as Hidden Markov Models (HMMs), which model the joint distribution of observations and labels, CRFs directly model the conditional probability distribution of the label sequence given the observed data. This approach allows CRFs to avoid making restrictive independence assumptions about the observations, making them more flexible and powerful in handling complex and structured data. CRFs are particularly well-suited for sequence labeling tasks because they can capture dependencies between neighboring labels.

A CRFs is defined on a graph , where represents the set of nodes corresponding to the random variables (labels), and denotes the edges representing dependencies between these variables. In the context of sequence labeling, CRFs are often defined on a linear chain, where each node corresponds to a label in the sequence, and edges connect adjacent labels, representing the dependencies between them [4].

Given an input sequence and a corresponding output sequence , a CRFs models the conditional probability of the output sequence given the input sequence as follows:

(2.6)w

where:

* are the parameters (weights) of the model.
* ) are feature functions that can depend on the current label ​, the previous label ​, the input sequence , and the position within the sequence.
* is the normalization factor (also known as the partition function).

This partition function sums over all possible label sequences , making exact inference computationally expensive for complex graph structures, though efficient algorithms exist for linear-chain CRFs.

2.10.1 Feature Function

Feature functions play a critical role in the framework of Conditional Random Fields (CRFs) and other machine learning models. They are the building blocks that allow these models to capture and represent the dependencies and characteristics of the input data, thereby enabling effective predictions. In CRFs, feature functions are used to measure the compatibility between the observation sequences and the label sequences, facilitating the computation of the conditional probability of a sequence of labels given the sequence of observations.

Feature functions are defined as binary or real-valued functions that can represent various attributes or patterns in the data. For instance, they can capture the presence of specific words, the position of words, or more complex dependencies such as the relationship between adjacent labels in a sequence. A feature function typically takes as input a label, the previous label, the observation sequence, and the position in the sequence, thereby encapsulating both local and contextual information. Mathematically, a feature function can be expressed as:

(2.7)w

)

where ​ and ​ represent the previous and current labels, respectively, denotes the entire observation sequence, and is the current position within the sequence. This formulation allows CRFs to leverage rich, expressive feature sets, capturing dependencies that go beyond the immediate input data.

The power of feature functions lies in their flexibility and expressiveness. They can be handcrafted based on domain knowledge, allowing experts to inject valuable insights into the model. Alternatively, they can be learned automatically from data, enabling the model to adapt to the unique characteristics of the task at hand. The weights associated with each feature function, often learned through training algorithms such as gradient descent, determine the influence of each feature on the final prediction. This enables CRFs to effectively balance the contributions of various features, optimizing the model's performance [9].

**2.11 Feature Extraction for POS Tagging**

Feature extraction is a critical step in Part-of-Speech (POS) tagging as it directly influences the performance and accuracy of the tagging model. In POS tagging, the objective is to assign the correct syntactic category (such as noun, verb, adjective, etc.) to each word in a given text. To achieve this, a set of relevant features must be identified and extracted from the text, which provides the necessary information for the model to make accurate predictions.

The features used for POS tagging typically include contextual and syntactic information surrounding each word. The following features are particularly important for capturing the nuances of the language:

* The Word Itself: The primary feature is the word itself, which directly provides clues about its likely POS category. For instance, words like "run" can be both nouns and verbs, and context plays a crucial role in determining the correct POS tag.
* POS Tag of the Word: Including the POS tag of the word when available (in the case of sequential tagging) helps refine predictions as the model progresses through the sentence.
* Previous Word: The word that immediately precedes the current word provides important contextual information. For example, if the previous word is an article like "the," it is likely that the current word is a noun.
* POS Tag of the Previous Word: The POS tag of the preceding word serves as a useful syntactic indicator, helping to establish a relationship between adjacent words. For example, if the previous word is tagged as an adjective, the current word might likely be a noun.
* Second Previous Word to the Given Word: Looking two words back in the sequence can reveal patterns that influence the current word's POS tag. For instance, a pattern such as "adjective + noun" may signal that the following word could be a verb in certain sentence structures.
* POS Tag of the Second Previous Word: This feature extends the context to two positions back, offering additional syntactic cues that can improve the tagging accuracy by considering broader contextual relationships.
* Next Word: The word immediately following the current word also provides valuable information. For example, if the next word is a verb, it may influence the tagging of the current word, particularly in ambiguous cases.
* POS Tag of the Next Word: Including the POS tag of the next word, when available, further refines the tagging process by incorporating forward-looking context into the decision-making process.

These features collectively allow the POS tagging model to leverage both local and broader context within the sentence, facilitating more accurate tagging. The careful selection and extraction of these features are essential in designing a robust POS tagging system that can effectively handle the complexities of the language being processed.

The Rawang language, a vibrant tapestry woven with unique alphabets, consonants, vowels, and tones, offers a glimpse into the rich cultural heritage of its people. This chapter has provided the essential foundation and theoretical framework for the journey of word segmentation and part-of-speech tagging. Understanding the Rawang writing system is key to unraveling the intricacies of the language and developing effective computational tools.

.

**CHAPTER 3**

**SYSTEM DESIGN AND IMPLEMENTATION**

This chapter details the development of word segmentation or tokenization and Part-of-Speech Tagging for Rawang language and provides a step-by-step explanation of the system design and implementation.

­

**3.1 System Design**

Start

End

Text Preprocessing

Word Segmentation

Rawang Text

BMM Algorithm

Word Corpus

POS Tagging

Labeled Corpus

Rawang Segmented Words with POS Tags

CRFs Model

**Figure 3.1 System Design**

The overall system design is illustrated in Figure 3.1. The process starts with Rawang text input, which undergoes preprocessing steps such as sentence segmentation, special character removal, lowercase conversion, and punctuation removal to standardize the text. Next, the text enters the word segmentation phase using the Bidirectional Matching (BDM) algorithm, which segments text into meaningful words by matching the longest character sequences against a predefined word corpus. The segmented text is then processed by a Conditional Random Fields (CRFs) model

for Part-of-Speech Tagging, assigning grammatical tags based on a labeled corpus. Finally, the system displays the words with their POS tags, offering a comprehensive analysis of the Rawang text. This design combines traditional text segmentation with machine learning to effectively manage the complexities of the Rawang language.

**3.2 System Flow of Bidirectional Matching in Word Segmentation**

Preprocessing

Rawang Text

Word Corpus

Are Forward and Backward Matches same?

Split with Space

Load Word Corpus

Backward Max Match

Forward Max Match

Compare the Length of Forward and Backward Matches

Choose Backward Match

Is Forward Match Longer?

Start

Choose Forward Match

Choose Forward or Backward Match

End

Is Valid Words?

Identity Valid Word or Phrase

Flatten into Format

Segmented Words

Append to Result

Yes

No

Yes

No

No

Yes

**Figure 3.2 System Flow of Bidirectional Matching in Word Segmentation**

Figure 3.2 illustrates the system flow for bidirectional matching in word segmentation of Rawang text. The process begins with the input of Rawang text, which is first subjected to preprocessing steps such as removing noise by eliminating special characters, converting text to lowercase, and removing punctuation. Following preprocessing, the text is split into a list of elements using spaces as delimiters, since Rawang language uses spaces to separate words or phrases. The system then loads a word corpus containing valid and unique Rawang words, which serves as a reference for identifying correct words or phrases in the text. Each element of the split list is checked against the word corpus to determine if it matches a valid word.

If an element matches a valid word, it is directly appended to the result. For elements that do not match valid words, the system treats them as potential phrases and applies the Forward Maximum Matching and Backward Maximum Matching algorithms. The Forward Maximum Matching scans the text from left to right, while the Backward Maximum Matching scans from right to left, each algorithm aiming to find the longest matching sequences from the corpus.

The results from the forward and backward matching are then compared. If the forward and backward matches are identical, either result is chosen. If the matches differ, the system compares the lengths of the longest sequences from both directions and selects the match with the longest sequence. Forward matches are selected if they are longer, otherwise, backward matches are chosen. The final segmented words are then flattened into a list, appended to the overall result, and displayed ready for subsequent POS Tagging.

**3.3 System Flow of Training and Testing Processes in CRFs Model**

Figure 3.3 illustrates the system flow for using Conditional Random Fields (CRFs) in Part-of-Speech (POS) tagging, divided into training and testing phases. During training, a labeled corpus is used for feature extraction, where various linguistic and contextual features are identified to help the model understand data patterns. The data is then used to train the CRF model, which learns the relationships between these features and the correct labels. Once trained, the model is saved for future use. In the testing phase, a separate labeled testing corpus undergoes the same feature extraction process to ensure consistency, and the pre-trained model applies these features to tag the test data, resulting in segmented words with their corresponding POS tags.

Start

Load Labeled Corpus

Training Corpus

Feature Extraction

Train using CRFs

Trained Model

Save Model

Start

Load Labeled Corpus

Testing Corpus

Feature Extraction

Load Trained Model

Label POS Tag Sets

Segmented Words with Tags

e.g. [word,tag]

End

End

**Testing**

**Training**

**3.3 System Flow of Training and Testing Processes in CRFs Model**

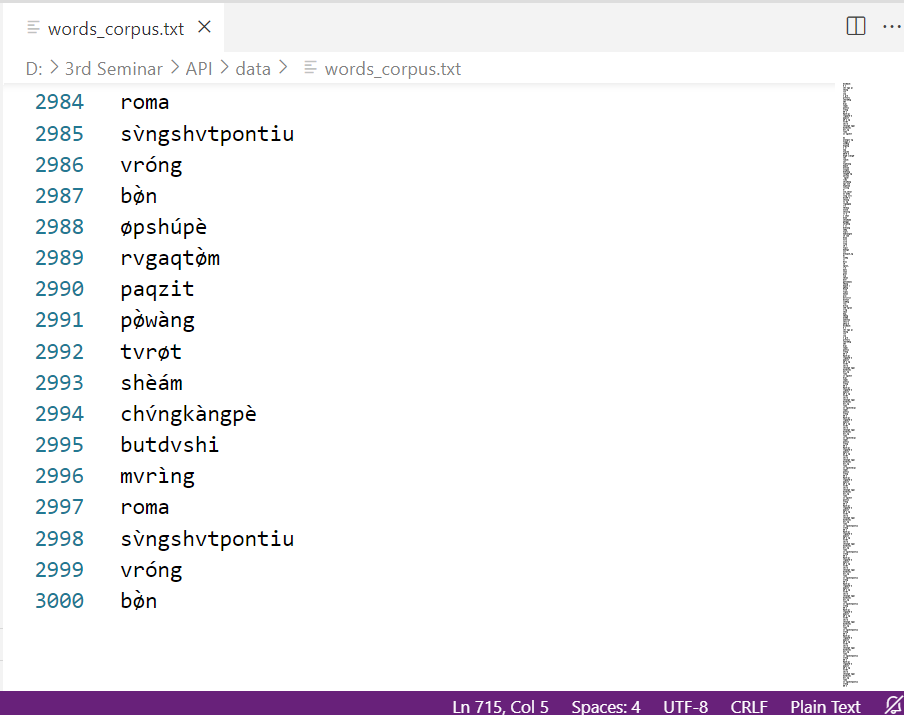
**3.4 Dataset**

The corpus for this system is compiled through meticulous manual data collection from two main sources, each offering valuable linguistic data. The first source is the Rawang Bible, specifically the books of Matthew, Mark, Luke, and John, chosen for their rich vocabularies and diverse linguistic expressions. These texts are valuable for capturing the nuanced features of the Rawang language, providing a strong foundation for the corpus. The second source involves direct interactions with native Rawang speakers, particularly older individuals who possess a deep and authentic understanding of the language. This system involves two distinct corpora: a unique word corpus for word segmentation and a grammar-labeled corpus for Part-of-Speech Tagging. The unique word corpus is designed to support accurate segmentation of Rawang words, while the grammar-labeled corpus provides essential part-of-speech annotations. The summary of these two corpora is shown in Table 3.1.

**Table 3. 1 Summary of Unique Words and Labeled Sentences Count by Bible Book**

|  |  |  |
| --- | --- | --- |
| **Bible Book** | **Unique Words Corpus** | **Labeled Sentences Corpus** |
| Matthew | 828 | 1022 |
| Mark | 750 | 940 |
| Luke | 515 | 880 |
| John | 680 | 858 |
| Other | 227 | 300 |
| **Total** | **3000** | **4000** |

3.4.1 Unique Word Corpus



**Figure 3.4 Unique Word Corpus**

The unique word corpus comprises nearly about 3,000 unique words, carefully selected to cover a wide range of linguistic structures. Each word is individually segmented and analyzed, providing a solid foundation for developing a reliable word segmentation model. The word corpus is stored in .txt format where each unique word is placed on a separate line, making it easily accessible for processing and analysis. An analysis of the corpus reveals several key statistics that underscore its complexity.

The average word length in the word corpus is 5.42 characters, reflecting the moderately intricate morphological structure characteristic of the Rawang language. The longest word in the dataset contains 17 characters, highlighting the language's capacity for compound words and morphological depth. The unique word corpus is shown in Figure 3.4.

3.4.2 Grammar Labeled Corpus

This system utilizes a grammar-labeled corpus containing nearly 4,000 sentences for Part-of-Speech Tagging. The labeled corpus is stored in .txt format, facilitates efficient processing and integration with the tagging system. The corpus comprises a total of 70,123 words, with an average sentence length of 19 words, which enhances its utility for comprehensive linguistic analysis. The grammar labeled corpus is shown in Figure 3.5.

****

**Figure 3.5 Grammar Labeled Corpus**

3.4.2.1 Rawang POS tag sets

A comprehensive tag set of 17 categories was developed through collaboration with the Rawang language experts. These tag sets were defined as an initial version and will be continuously refined to improve its accuracy and comprehensiveness. The finalized Rawang POS tag sets of this system are shown in Table 3.2.

**Table 3. 2 Rawang POS Tag Sets**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Grammar** | **POS Tag Sets** | **Description** |
| 1 | Noun | NN | words that identify people, places, things, or ideas. |
| 2 | Pronoun | PRON | words that take the place of nouns in a sentence. |
| 3 | Verb | V | words that denote actions, states, or occurrences. |
| 4 | Adjective | ADJ | words that describe or modify nouns, providing more detail about them. |
| 5 | Preposition | PREP | words that establish relationships between other words in a sentence, typically indicating location, time, direction, or manner. |
| 6 | Adverb | ADV | words that modify verbs, adjectives, or other adverbs. |
| 7 | Auxiliary Verb | AUX | words that confirm the work that will do. |
| 8 | Conjunction | CONJ | words that connect words, phrases, or clauses within a sentence. |
| 9 | Interjection | INT | words or phrases that express strong emotions or exclamations. |
| 10 | Interrogative | IR | words that are used to ask questions and elicit information. |
| 11 | Number | NUM | words that indicate numerical values. |
| 12 | Number Text | NT | words that express numerical values in written form. |
| 13 | Particle | P | small, often uninflected words that do not fit neatly into other grammatical categories. |
| 14 | Case | C | words that express someone’s possessive. |
| 15 | Quantity | Q | words that indicate quantity. |
| 16 | Demonstrator | DM | words that are used to indicate or point out specific items or entities in relation to the speaker or listener. |
| 17 | Chronological Verb | CV | words that indicate tense. |

**3.5 System Implementation**

This section outlines the implementation of word segmentation and Part-of-Speech Tagging for the Rawang language. Word segmentation involves dividing a sequence of text into individual words or meaningful units. Following this, Part-of-Speech Tagging assigns grammatical labels to each segmented word, reflecting the syntactic roles or categories that the words play within a sentence.

The system is implemented using Python for the core processing tasks. Flask is employed for server (back-end) hosting, facilitating the management of web requests and responses. The Graphical User Interface (front-end) is developed using Flutter, a cross-platform framework that allows for the creation of applications for android, iOS, web, desktop, Windows, and macOS.

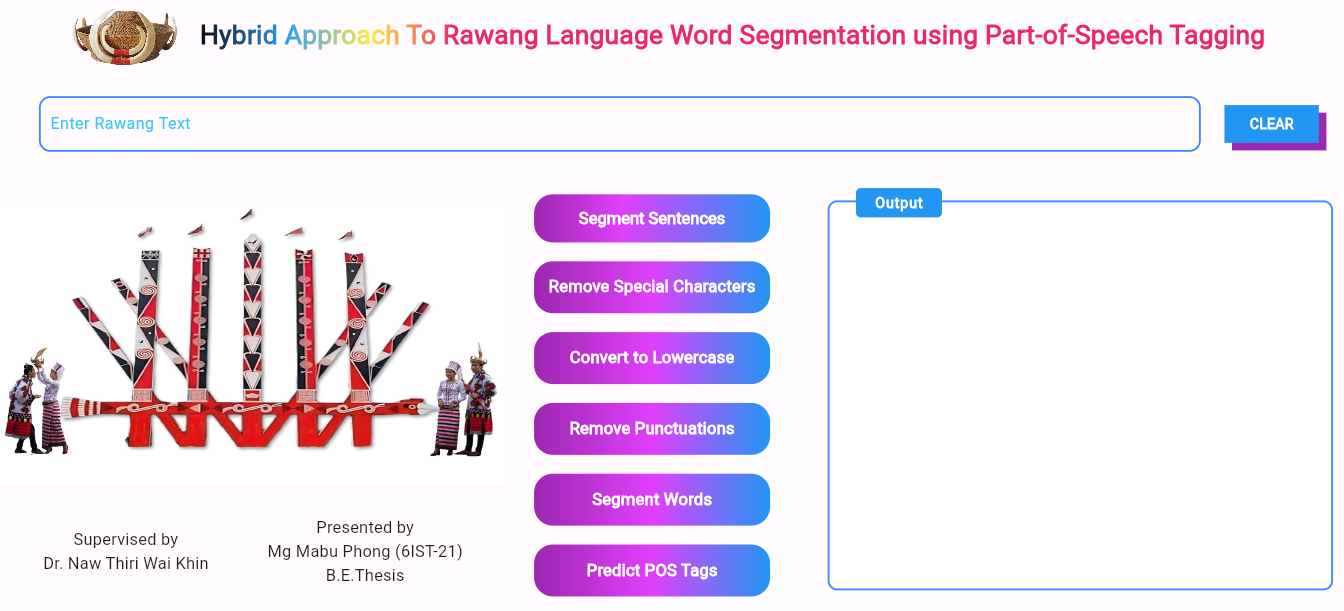
**Figure 3.6 User input of the System**

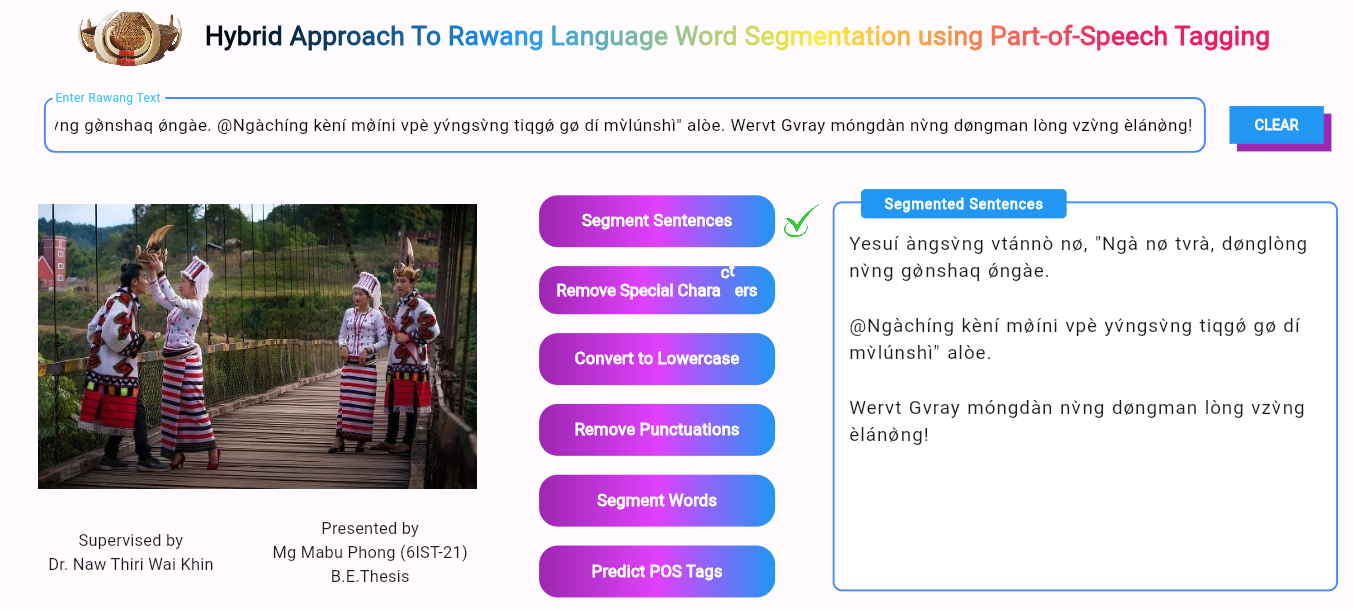
Figure 3.6 is the user input of the system. To segment and tag pos-labels, user must type Rawang text. This system can accept multiple sentences for processing, allowing user to input extensive text for segmentation and tagging.

3.5.1 Text Preprocessing

Before advancing to word segmentation and Part-of-Speech Tagging, text preprocessing is carried out on the input from the text box. This process includes segmenting sentences, removing special characters, converting text to lowercase, and eliminating punctuation. This clean and well-structured input enables the model to learn more effectively, leading to more reliable and interpretable results in the final analysis.

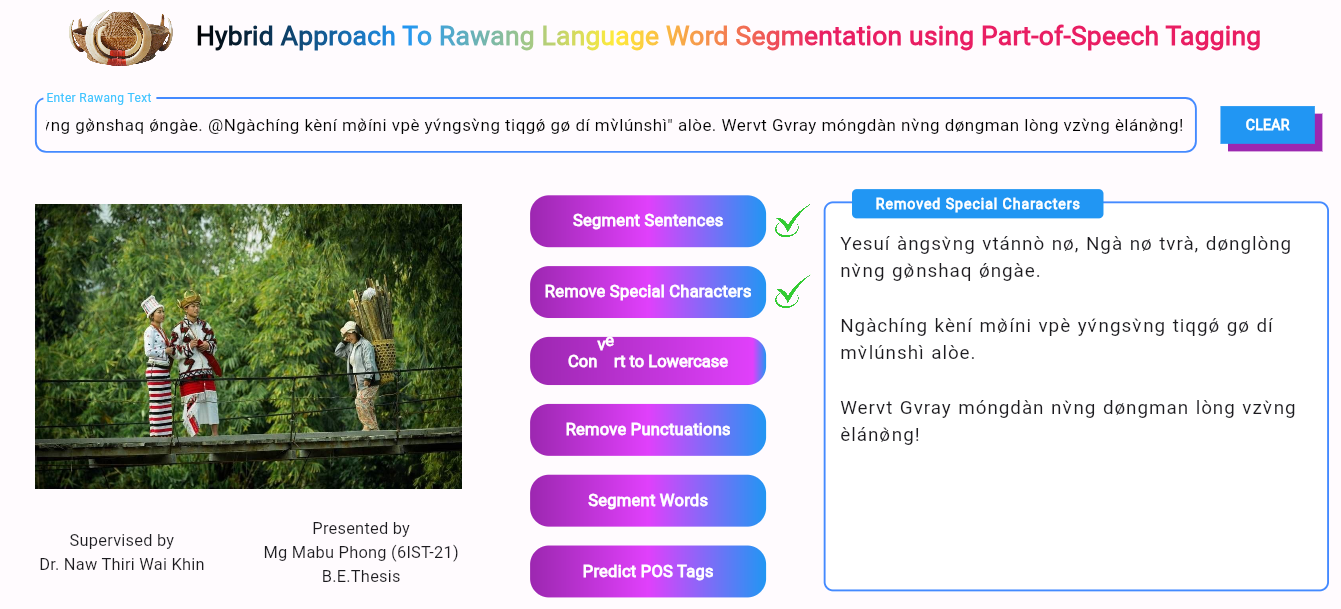
3.5.1.1 Sentence segmentation

In the Rawang language, the period (.), question mark (?), exclamation (!), comma (,), semicolon (;), colon (:) and dash (-) are recognized as sentence delimiters. Therefore, the system segments text into individual sentences by identifying these punctuation marks, ensuring that each sentence is displayed on a separate line as shown in Figure 3.7.

**Figure 3.7 Sentence Segmentation**

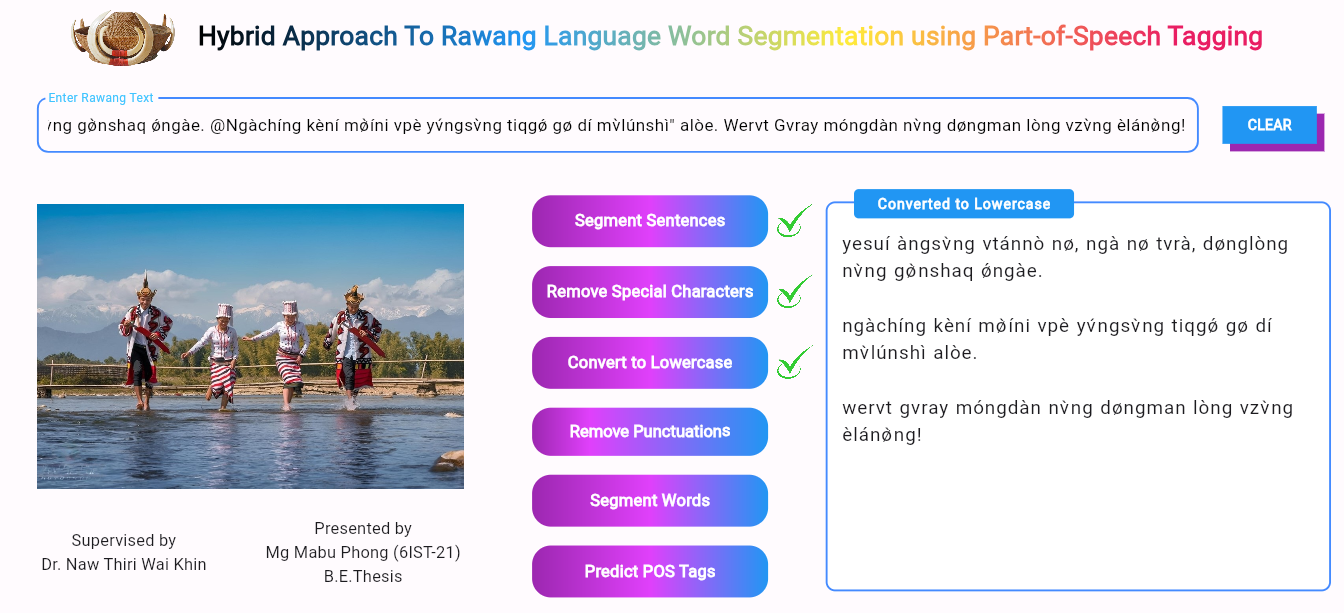
3.5.1.2 Removing special character

This process involves filtering out non-alphanumeric symbols that do not contribute to the meaning of the text, such as hashtags, asterisks, or other special characters. The result of removing special characters is demonstrated in Figure 3.8.

**Figure 3.8 Removing Special Characters**

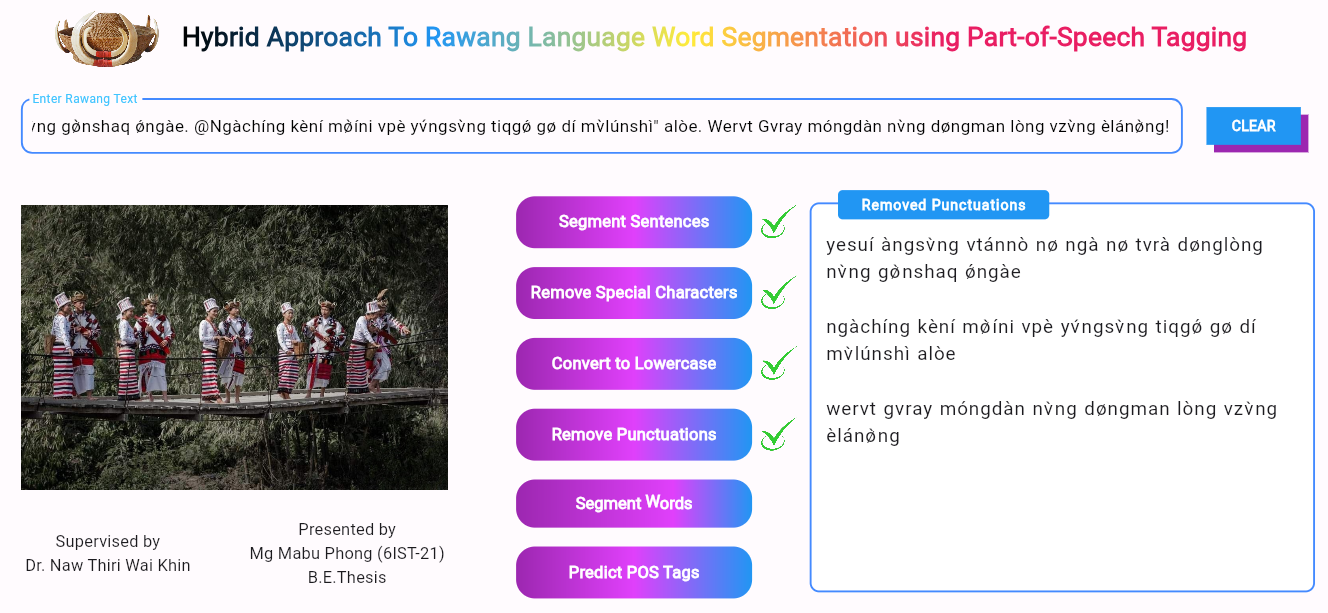
3.5.1.3 Converting to lowercase

This phase removes variations introduced by uppercase letters, standardizing the text to a uniform lowercase format. This uniformity allows the system to process the text more consistently, reducing potential discrepancies caused by case differences. The outcome of this conversion is shown in Figure 3.9.

**Figure 3.9 Converting to Lowercase**

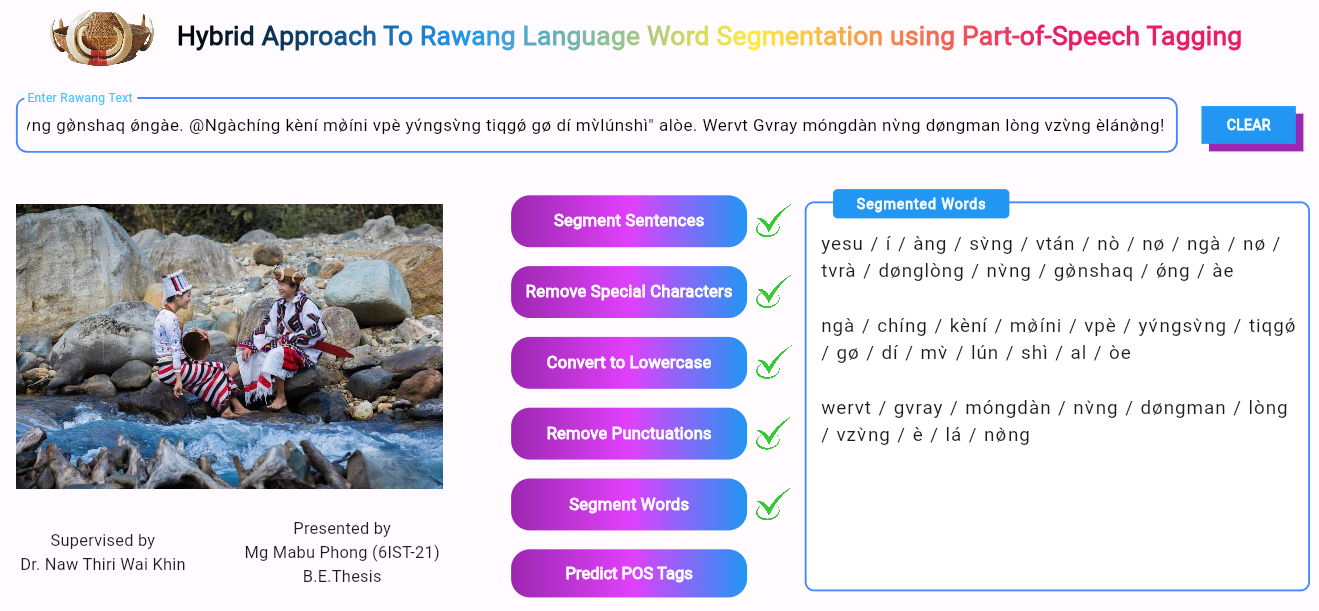
3.5.1.4 Removing Punctuations

This preprocessing involves stripping out punctuation marks, such as commas, question mark, periods, colon, semicolon, dash and exclamation points, the system can focus solely on the words themselves, reducing potential noise in the data. Figure 3.10 shows the result of removing punctuation on the text.

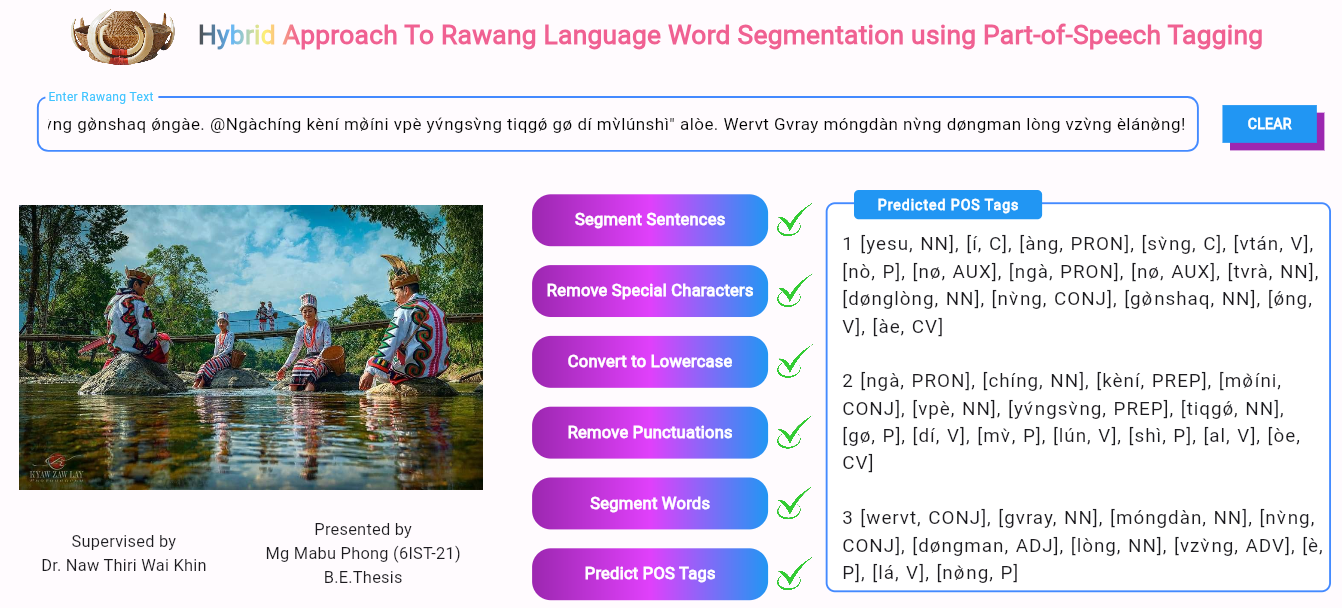
**Figure 3.10 Removing Punctuations**

3.5.2 Word Segmentation

After text preprocessing, the text is forwarded to the Python backend server, where the word segmentation process takes place. The server utilizes the Bidirectional Maximum Matching Algorithm to efficiently segment the text into individual words. Once the segmentation is complete, the results are sent back to the front end and displayed each word with ‘/’ delimiter. This seamless interaction between the backend and frontend ensures that the segmented words are accurately processed and promptly returned to the user. The results after word segmentation is shown in Figure 3.11.

**Figure 3.11 Word Segmentation**

3.5.3 Part-of-Speech Tagging

 **Figure 3.12 Part-of-Speech Tagging**

After word segmentation, the segmented words are sent to the backend server for Part-of-Speech (POS) tagging. The server utilizes a trained Conditional Random Fields (CRFs) model, based on an annotated corpus, to predict the appropriate POS tags for each word. Once the tagging process is complete, the predicted POS tags are returned to the frontend, where they are displayed alongside the segmented text. The final output, including the segmented words and their corresponding POS tags, is shown in Figure 3.12.

**3.6 Part-of-Speech Tagging Accuracy**

In this section, the accuracy value of part-of-speech tagging is described.

3.6.1 Hyperparameter Tuning

To improve the performance of the CRFs model for Part-of-Speech Tagging, hyperparameter tuning is performed by adjusting the values of c1 (L1 regularization) and c2 (L2 regularization). The tuning process involves testing different combinations of these parameters to identify the optimal configuration.

**Initially, the parameter values are set to {'c1': [0.1, 1, 10], 'c2': [0.1, 1, 10]}. The resulting accuracy is shown in Table 3.3.**

**Table 3.3 Part-of-Speech Tagging Accuracy**

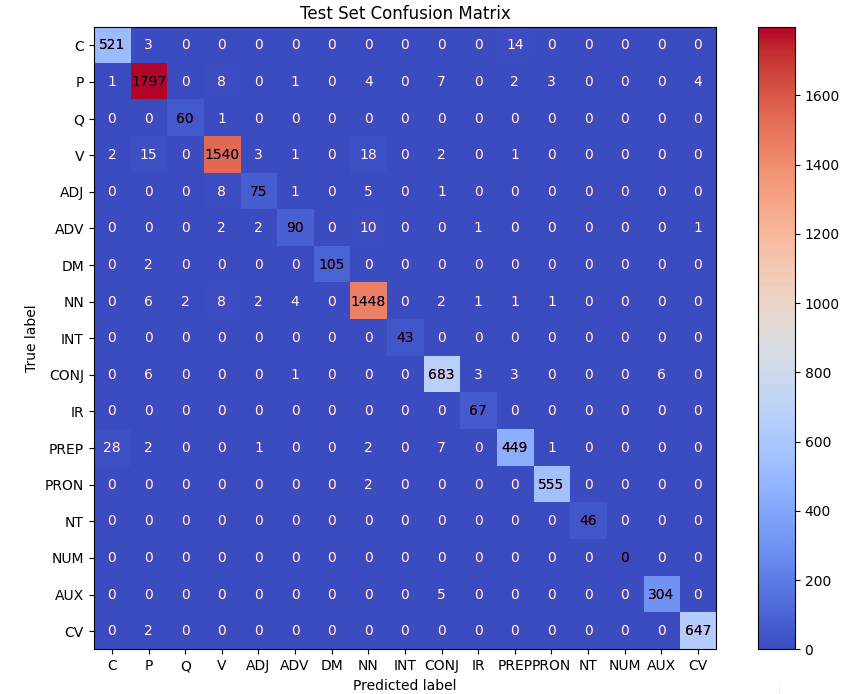
|  |  |  |  |
| --- | --- | --- | --- |
|  | **Training** | **Testing** | **Accuracy** |
| Split Data | 60% | 40% | **94%** |
| 70% | 30% | **95%** |
| 80% | 20% | **96%** |

When the parameter values are set to **{'c1': [0.01, 0.1, 1], 'c2': [0.01, 0.1, 1]}, the accuracy is obtained as shown Table 3.4.**

**Table 3.4 Part-of-Speech Tagging Accuracy**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Training** | **Testing** | **Accuracy** |
| Split Data | 60% | 40% | **96%** |
| 70% | 30% | **97%** |
| 80% | 20% | **98%** |

After adjusting the c1 and c1 values to smaller ranges, the model's accuracy improves significantly. This highlights the importance of fine-tuning hyperparameters, as even slight changes can lead to substantial gains in model performance. The confusion matrix for the test set, trained on 80% of the data and tested on the remaining 20% using {'c1': [0.01, 0.1, 1], 'c2': [0.01, 0.1, 1]}, is displayed in the Figure 3.13.

****

**Figure 3.13 Test Set Confusion Matrix**

In summary, this chapter has established the groundwork for a robust system to segment and tag Rawang words by blending traditional linguistic knowledge with advanced computational techniques. The use of carefully curated datasets, along with the implementation of effective preprocessing, segmentation, and tagging algorithms, provides a robust foundation for further research and development in Rawang language processing.

**CHAPTER 4**

**CONCLUSION**

**4.1 Conclusion**

This system developed a hybrid approach to Rawang language word segmentation and Part-of-Speech Tagging, utilizing advanced preprocessing techniques and a trained Conditional Random Fields (CRFs) model. The system efficiently processes Rawang text, supporting multiple sentences and applying steps like sentence segmentation, special character removal, conversion to lowercase, and punctuation removal. The use of the Maximum Matching algorithm and the CRFs model ensure accurate segmentation and POS tagging tailored to the Rawang language.

The user-friendly interface simplifies these complex tasks, making the system accessible to both novice and expert users. This work contributes to the digital preservation and study of the Rawang language, providing a practical tool for researchers and language enthusiasts. Future enhancements could expand the system's capabilities and its integration with broader language processing tools.

**4.2 Benefits**

The hybrid approach presented in this system offers substantial advantages for Rawang language processing. By combining the strengths of Maximum Matching (MM) and a trained CRFs model, the system significantly enhances accuracy in both word segmentation and Part-of-Speech Tagging. This innovative approach effectively tackles the complexities of the Rawang language, producing more precise and reliable linguistic analyses.

Comprehensive preprocessing steps, including sentence segmentation, character normalization, and punctuation removal, ensure that the input text is optimally prepared for processing. This meticulous preparation contributes to the overall quality and efficiency of the system.

The user-friendly interface democratizes access to advanced language processing tools, empowering both novice and expert users. This accessibility fosters wider adoption and facilitates in-depth exploration of Rawang language. By providing

a practical tool for researchers and language enthusiasts, the system supports the preservation and development of Rawang linguistic resources. The modular design of the system lays a strong foundation for future expansions. Integration with broader language processing platforms and the incorporation of additional linguistic features are promising avenues for future research.

Additionally, the system will be made available online for free, ensuring that everyone can access and learn from the research. This open-access approach enhances the educational value of the work, allowing a global audience to benefit from the advancements in Rawang language processing.

**4.3 Limitations**

While the developed system offers valuable contributions to NLP for the Rawang language, it is important to acknowledge its limitations.

* **Language-Specific Focus**: The system is specifically designed for the Rawang language, limiting its applicability to other languages due to the unique linguistic characteristics and challenges inherent to Rawang.
* **Limited Dataset**: The system's performance is restricted by the small and homogeneous dataset, primarily comprising Rawang Bible texts from the books of Matthew, Mark, Luke, and John. This limited data impacts the system's ability to generalize across diverse contexts.
* **Accuracy Limitations**: The system cannot accurately tag every Rawang word because the training corpus is derived from specific sections of the Rawang Bible. This focused dataset limits the system's ability to handle words and expressions that fall outside these texts, reducing its overall tagging accuracy.
* **Dependency on Manual Data Collection**: The current reliance on manual data collection and annotation is both time-consuming and labor-intensive. Implementing automated data collection and annotation methods could enhance the system’s efficiency and scalability.
* **Impact of Incorrect Segmentation**: Incorrectly segmenting words can reduce the accuracy of POS tagging, as the segmentation process directly influences the tagging results. Errors in segmentation can lead to incorrect or missed POS tags, thereby affecting the overall performance of the system.
* **Unstable Rawang Grammar**: The Rawang language lacks a well-documented and standardized grammar, making it difficult to develop accurate and reliable Natural Language Processing models. This instability adds to the complexity of linguistic analysis and system development.

**4.4 Further Extension**

The future development of this system presents numerous opportunities to enhance its capabilities and broaden its scope. By adapting cutting-edge advancements in Natural Language Processing and integrating cross-platform support and user feedback, expanding the system's linguistic and functional range, the potential for improved performance and wider applicability is significant. Key areas of focus for future extensions include:

* **Integration of Advanced Machine Learning Techniques**: Enhancing the system's accuracy and robustness in word segmentation and Part-of-Speech Tagging can be achieved by incorporating advanced machine learning techniques, such as transformer models. These models have shown remarkable success in various NLP tasks and could significantly improve the performance of the Rawang language processing.
* **Incorporation of Complex Linguistic Features**: Expanding the system to include syntactic parsing and named entity recognition would deepen its understanding of the Rawang language's structure. Analyzing syntactic relationships and identifying key entities will provide more nuanced insights into the language's usage and grammar.
* **Expansion of the Dataset**: To train a more robust and versatile NLP model, it is essential to develop a comprehensive dataset that includes a diverse range of Rawang texts and dialects. This expansion will help capture the linguistic variability within the Rawang community.
* **Feedback and Evaluation**: Actively involving the Rawang-speaking community in the evaluation process through feedback is crucial for refining the system. By gathering user input on accuracy, usability, and overall effectiveness, specific areas for improvement can be identified and addressed, leading to a more user-friendly and reliable solution.
* **Collaboration with Language Processing Platforms**: Building partnerships with other NLP platforms and research initiatives can promote collaborative advancements and cross-linguistic insights. Sharing expertise, resources, and best practices with the wider NLP community will accelerate the development of innovative language processing solutions.
* **Engagement and Education of the Rawang Public**: Introducing the system to the Rawang-speaking community through workshops, demonstrations, and educational resources will foster greater understanding and adoption. Educating the public on the system's benefits can encourage its use in language preservation, education, and research initiatives.

**REFERENCES**

[1] David Sangdong, “Rawang-English-Burmese Dictionary,” in Rawang-English-Burmese Dictionary, Randy J. LaPolla and David Sangdong, Myitkyina, Kachin, Myanmar, 2015, pp. 1-480.

[2] Isaiah Yøntv̀ng, “Rvwàng Grammar,” in Rvwàng Grammar & Word Order, vol. 1, Isaiah Yøntv̀ng, Chiang Mai, Thailand, 2004, pp. 6-222.

[3] H. Liu and M. Zhou, “Bidirectional maximum matching for Chinese word segmentation,” in Proceedings of the 20th International Conference on Computational Linguistics, 1st ed. Geneva, Switzerland: International Committee on Computational Linguistics, 2004, pp. 448-454. [Online]. Available: http://www.aclweb.org/anthology/C04-1082

[4] C. Sutton and A. McCallum, “An Introduction to Conditional Random Fields for Relational Learning,” in Introduction to Statistical Relational Learning, L. Getoor and [5] B. Taskar, Eds., 1st ed. Cambridge, MA, USA: MIT Press, 2007, pp. 2-19. [Online]. Available: https://people.cs.umass.edu/~mccallum

[5] Y. Chen and T. K. Ho, “Bidirectional maximum matching for word segmentation and its applications,” presented at the International Conference on Computational Linguistics (COLING), Taipei, Taiwan, Dec. 17–21, 2002.

[6] P. V. Eugene and R. R. Shavlik, “Improving part-of-speech tagging with a hybrid model,” presented at the 17th International Conference on Computational Linguistics (COLING), Taipei, Taiwan, Aug. 23–27, 1998.

[7] Hpung Sarep, “A Study of the Morphology of Verbs and Nouns of The Sinwal Dialect of the Rawang Language,” presented at Payap University, Chiangmai, Thailand, May, 1995.

[8] X. Huang and S. H. Chen, “A hybrid approach for Chinese word segmentation,” presented at the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), Brussels, Belgium, Oct. 31 – Nov. 4, 2018.

[9] J. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” presented at the 18th International Conference on Machine Learning (ICML), Williamstown, MA, USA, June 28–July 1, 2001.

[10] Manning, Christopher D., and Hinrich Schütze. “Foundations of Statistical Natural Language Processing”. MIT Press, 1999.

https://mitpress.mit.edu/9780262133609/foundations-of-statistical-natural-language-processing/ (accessed March 21, 2024).

[11] Saneshank, "CRFs\_POS\_tagging," GitHub,

https://github.com/saneshashank/CRF\_POS\_tagging (accessed April 6, 2024).