ISSN 2007-9737

Part-of-Speech Tagging for Mizo Language Using Conditional Random Field

Morrel V. L. Nunsanga[1], Partha Pakray[2], C. Lallawmsanga[1], L. Lolit Kumar Singh[3]

[1,3] Mizoram University,
Department of Information Technology,
India

[2] National Institute of Technology Silchar,
Department of Computer Science and Engineering,
India

[3] Mizoram University,
Department of Electronics and Communication Engineering,
India

morrelhmar@mzu.edu.in, pakraypartha@gmail.com, lomsanga@rediffmail.com,
llksingh@yahoo.co.in

Abstract. Part of speech (POS) tagging assigns a class or tag to each token in a sentence. The tag allocated to a word is mainly its part of speech or any other class of interest. Several applications of Natural Language Processing (NLP) require it as a prerequisite. The development of part-of-speech tagging for the under-resourced Mizo language is presented in this study, which makes use of a stochastic model known as Conditional Random Field (CRF). The CRF is a discriminative probabilistic classifier that considers both the context of a given word and the tag transition probabilities in the training dataset. A corpus of approximately 30,000 words was collected and manually annotated with the proposed tagset for system evaluation. On various sizes of training and test sets, the tagger achieved 89.46 % accuracy, 89.3 % F1-score, 89.42 % precision, and 89.48 % recall.

Keywords. Mizo POS tagging, conditional random field, Mizo part of speech tagger, computational linguistics.

1 Introduction

The task of using computer programs to process natural languages in their written or oral forms in order to extract meaningful information is known as natural language processing (NLP). It is a sub-field of Artificial Intelligence (AI) that aims to facilize natural communication with computers. The rapid advance in the field of NLP and more and more of its usage being integrated into our daily lives compels one to harness its promising potential.

For many natural language processing tasks, a reliable POS tagger is a prerequisite. A POS tagger accepts a sequence of text in a particular language as input, and the tagger assigns the appropriate tag to each word in the sequence. The performance of a POS tagger directly determines the quality and reliability of subsequent phases of NLP tasks. Different POS tagging models, therefore, need to be studied and evaluated to determine their suitability for a language under consideration.

Resolving ambiguity is a major challenge of POS tagging since, by nature, most words have multiple senses. A completely correct POS tagging would require other information such as syntax, semantics, and world knowledge. Since the only information we have at the POS tagging phase is word-level information such as morphological information, a POS tagging cannot be expected to be 100% correct. Till date, there is currently no known solution that can answer the part of speech tagging problem with 100% accuracy in any language, including English.

However, a high degree of accuracy can be obtained, which can be used for practical purposes. Although POS tagging is not effective in and of itself, it is widely acknowledged as the first step in comprehending a natural language. Many natural language processing activities and applications, including speech synthesis and recognition, parsing, machine translation, and information extraction, rely significantly on it.

Mizo language is classified under the Tibeto-Burman language family. It is the primary and most widely spoken language in Mizoram, a state in northeastern India. Aside from Mizoram's mainland, this language is also spoken in the surrounding states such as Tripura, Assam, Manipur, Meghalaya, Nagaland, and lesser parts of Myanmar and Bangladesh.

There was no text writing system for the Mizo language until the arrival of two pioneer Christian missionaries, James Herbert Lorrain (Pu Buanga) and Dr. Frederic William Savidge (Sap Upa), in Mizoram in 1894 [17]. The two missionaries started the work on developing the Mizo alphabet, and it was completed on the 1st of April, 1894. Before crafting the Mizo alphabet, a thorough comparison was made to determine which Indian scripts, such as Hindi and Bengali, and the Roman script, should be used as the foundation of the alphabet. They believed that the Roman script was more appropriate for the Mizo language. In addition, the two missionaries developed the first Lushai grammar and dictionary, which served as the foundation for the Mizo language and literature to be developed in the following decades.

Mizo language is a tonal language in which the tone, pitch, and contour of the syllables can change the meanings of a word. Its tonal character is a hurdle in computational linguistics because there are no universally accepted and widely acknowledged tonal symbols to represent all the different tones in the language. Certain publishers recommended the use of diacritics (á, à, é, è, ó, ù) to denote the tones and intonations used in their publications, despite the fact that they were not standard. Mizo language is still in its infancy in terms of language processing applications. Reliable resources need to be developed and more efforts need to be put for research works so that the language can be integrated with modern NLP applications. This work is an attempt in that direction.

The sections that follow are organized as follows: The relevant works are highlighted in Section 2. Section 3 presents the system description, including the Conditional Random Field (CRF) model. Section 4 contains the implementation and analysis of the results, while Section 5 contains the conclusion to this research work.

2 Related Works

Numbers of researchers have given efforts for the development of part of speech tagging for various languages. Despite this, only a few research studies on POS tagging for the Mizo language were found. This section highlights some of the related POS tagging approaches for different languages.

Using the CRF model and the hidden Markov model, Aswathi et al. [1] presented a paper on POS tagging and chunking. The tagger was based on the combination of the stochastic model and the rule-based approach. The main idea was to perform the initial tagging with the TnT (i.e., the second-order HMM) and then apply the proposed set of rules to handle the errors generated by the tagger. The CRF-based tagger was also developed in this research work. It was observed that the TnT tagger outperformed the CRF-based tagger, and the performance of the TnT tagger was further improved and yielded F-measure tp 80.74 with the transformation processing technique.

Deskmuk et al. [2] presented POS tagger for Marathi language using Bi-LSTM (Bidirectional long short-term memory) and deep learning model. The results were compared with different machine learning techniques. The deep learning model and Bi-LSTM yielded better accuracy than most of the machine learning methods. 85% accuracy was achieved for both the deep learning model and the Conditional random field model. The best accuracy was obtained with the Bi-LSTM method (97%).

A part-of-speech tagger for Manipuri based on Conditional Random Field and SVM was presented by Singh et al. [3]. A corpus was built from various sources of text dataset, which was manually annotated with 26 tags.

The tagger employed a variety of contextual and orthographic features at the word level. The proposed system was trained on a manually tagged corpus of 39449 words, tested on 8762 tokens, and an accuracy of 72.04% was achieved.

Using Conditional Random Field as a language model, Pandian et al. [4] presented POS tagging and chunking for the Tamil language. A morphological analyzer was utilized since Tamil is a language with a diverse morphology. The model was trained on 39000 sentences and observed the performance of the tagger with three different test sets. The authors reportedly achieved an accuracy of 89.18%.

Using CRF and Support Vector Machines (SVM), Outahajala et al. [5] performed the first part-of-speech tagger for the Amazighe language. Around 20000 manually tagged tokens were used for the experiment. An open-sourced CRF++ was used in the experiment and claimed to have achieved an accuracy of 88.66% using the CRF model and 88.27% using the SVM model.

For the Meitei Mayek Manipuri language, a combination of transliteration and a CRF-based POS tagger was developed by Nongmeikapam et al. [6]. Conditional Random Field (CRF) was used to assign the parts of speech tag in the Bengali Script Manipuri text, which was then transliterated into Meitei Mayek. In the experiment, a corpus of 30000 words was divided into 24000 and 6000 words for training and testing purposes, respectively. The authors claimed to have achieved an accuracy of 86.04 % using the CRF++ 0.53 package.

Kumar et al. [7] proposed a CRF model and second-order HMM-based Kannada language part-of-speech tagging system. The systems were trained on a dataset containing 51,269 tokens and then tested on a dataset containing 2932 tokens. The corpus was taken from the EMILLE corpus. The authors claimed to achieve 79.9% accuracy using the HMM-based tagger and 84.58% accuracy using the CRF-based tagger, respectively.

Ojha et al. [8] presented the training and evaluation result of Conditional Random Field-based part-of-speech tagger and Support Vector Machin(SVM)-based POS tagger on Hindi, Odia, and Bhojpuri languages.

The experiment used a training dataset of 90,000 words and a test dataset of 2000 words. Data for the experiment was extracted from the Indian Language Corpora Initiative (ILCI), and the BIS annotation scheme was used. The accuracy obtained ranges from 82-86.7% for the CRF model and 88-93.7% for the SVM model. In comparison to SVM, the study reported that languages with more variations are better suited for CRF.

Ghosh et al. [9] performed POS tagging using Conditional Random Field on a code-mixed social media text that included English, Hindi, Tamil, and Bengali. A conditional random field was used to develop the final system after starting with the Stanford Part of Speech tagger. A variety of pre-processing and post-processing modules was implemented in order to enhance the system's performance. A CRF++ toolkit was utilized for implementing the model. They claimed to have achieved an accuracy of 75.22 % when dealing with the data in Bengali-English code-mixed.

Zeroual et al.[10] conducted a detailed examination of the tagset for the Arabic language and produced a hierarchical level for the language's tagset. The study's primary purpose was to enhance the performance of the taggers built for the Arabic language by providing the finest tagset feasible for the language that covered its complicated morphological structure. It was demonstrated experimentally that the proposed tagset produced more precise and accurate results. The usability of the proposed tagset was assessed with the help of the Treetagger.

POS tagger using SVMTool for under-resourced Setswana African Language has been discussed in [11]. The model was evaluated with 60% of the corpus as training data and 40% of the corpus as testing data. By applying different strategies, the highest accuracy achieved with the model was 92.16%.

Part-of-speech tagging related to the Mizo language was discussed in [d,e]. These are the only few publications on Mizo part-of-speech tagging that we are aware of, to the best of our knowledge. The main objective of this study [d] was to lay the foundation of POS tagging for Mizo. In this study, a tagset consisting of 26 tags and a Mizo-to-English dictionary containing 26,407 patterns for the Mizo language POS tagging system was presented.

Lawmsanga et al. [c] discussed the Mizo language's unique features as well as the challenges of the tagging system in Mizo.

3 System Description

The development of the proposed system in various phases such as data collection, pre-processing, tokenization, tagset, corpus creation, and the CRF models are discussed in this section.

3.1 Data Collection

Texts used for the creation of custom corpus are collected from 'Vanglaini', the most widespread daily newspaper in the state. Care is taken so that sentences in the text conform as close as possible to the grammar rules.

The raw texts are chosen from domains such as sports, politics, news, music, health, religion, etc., to capture the possible occurrence of different use cases of a word in various domains. The collection amounted to 30647 words in 968 sentences (An average of 31.6 words/sentence).

3.2 Preprocessing

Further processing of the collected raw text is required in order to leverage inconsistent writing styles of different contributors. Most of them are a result of ignorance of grammar in general. e.g inconsistency in some compound words is very common wherein the same compound word is written as spaced compound noun, solid compound noun (without any space in between) or as a hyphenated compound noun. Available grammar books [13,14,15,16] as well as blog posts of well-known experts are referred for making necessary corrections.

3.3 Tagset

A tagset is a collection of tags or grammatical classes to which each token in the test dataset has to be classified. When creating a tagset, it is necessary to include overt morphological differences in the language. Table 1 shows a tagset for Mizo language, consisting of 47 tags, that was created by modifying the tagset proposed by [19], which was utilized to annotate the collected corpus.

3.4 Tokenization

Tokenization is the process whereby raw text is further split into smaller chunks of tokens suitable for further processing. For this study, the phrases are tokenized into words separated by exactly one space. Punctuations and symbols are treated as separate words and are thus labeled accordingly. Since the corpus needs to be processed sentence-wise, tokenized words are grouped into sentences. Each sentence is separated by a newline character.

3.5 Creation of Custom Mizo Corpus

Mizo language does not have a publicly available tagged corpus, so it is necessary to create a new one. A POS tagged corpus is created from the tokenized text by manually tagging each token or word with its appropriate tag by putting the '/' symbol between the word and its corresponding tag. A summary of the tagged corpus is shown in table 2. A sentence in the tagged corpus would look like the following:

Lungphum/CMN *phumte*/CMN *chu*/AT *cheng*/CMN *nuai*/CMN *3334*/CD *senga*/SPRB din/VB tur/RB a/PSP *ni*/VB ./.

January/ET *15-ah*/RBT *sikul*/CMN *kal*/VB *theih*/RB *beisei*/VB ./.

3.6 Specification of Features

Attributes for CRF feature functions need to be fed to the model, which is basically a specification of the context of a given word in the sentence. The features selected for the experiment are given in table 3. The CRF model uses these features from the training set to build feature functions.

3.7 Conditional Random Fields

Let y be a vector that represents a label sequence and x be the corresponding vector that represents the observation sequence. Given two variables, x, and y, the CRF directly models p(y|x), the

Table 1. POS tagset for Mizo language

Tags	Description
PPN	Proper Noun
CMN	Common Noun
ABN	Abstract Noun
PSP	Personal Pronoun
POP	Possessive Pronoun
RLP	Relative Pronoun
IP	Interrogative Pronoun
MP	Demonstrative Pronoun
JJ	Adjective base form
MJJ	Demonstrative Adjective
DJJ	Double Adjective
IJJ	Interrogative Adjective
NJJ	Nounal Adjective
CJJ	Comparative Adjective
SJJ	Superlative Adjective
VB	Verb base form
NVB	Nounal Verb
DVB	Double Verb
RB	Adverb base form
DRB	Double Adverb
MRB	Demonstrative Adverb
PPT	Postposition
CC	Coordinating Conjunction
UH	Interjection
PT	Particles
SYM	Symbol
,	Comma
.	Fullstop
:	Colon
;	Semi colon
?	Question mark
(	Open bracket
)	Close bracket
QM	Quotation Mark
CD	Cardinal number
NG	Negation
ET	Date
RBP	Adverb of Place
RBT	Adverb of Time
SF	Suffix
AT	Article
RBM	Adverb of Manner
FW	Foreign Word
CRB	Comparative Adverb
SRB	Superlative Adverb
VBN	Verbal Noun

conditional distribution of y given x. Lafferty et al. [12] pioneered the use of Conditional Random Fields for data labeling and segmentation. According to [12], the distribution of output vector y given x (the two vectors have the same length) is a product of potential functions described by the following expression:

$$exp(\sum_m \lambda_m f_m(y_{i-1}, y_i, \boldsymbol{x}, i) + \sum_s \mu_s g_s(y_i, \boldsymbol{x}, i)), \quad (1)$$

where the first part of the eq. 1 $f_m(y_{i-1}, y_i, x, i)$ is a set of feature functions based on the whole observation sequence considering the output variables at positions *i* and *i-1*. The second part of eq. 1 is a state feature function whose input is the label at position i and the sequence of observation denoted by $g_k(y_i, x, i)$.

The feature functions are represented by a set of real-valued functions $g_s(y_i, x, i)$ It can be any real-valued positive function which reflect some characteristic of the training data. Features are selected such that they reflect the CRF model considered.

Simplifying the notation in Eq. 1 by writing:

$$g(y_i, x, i) = g(y_{i-1}, y_i, x, i),$$

and

$$F_m(y, x) = \sum_{i=1}^{n} f_m(y_{i-1}, y_i, x, i), \quad (2)$$

where each function $f_m(y_{i-1}, y_i, x, i)$ can be a state function $g(y_{i-1}, y_i, x, i)$ or a transition function $f(y_{i-1}, y_i, x, i)$ and *n* is length of observation sequence.

The probability of output sequence y given x is given by:

$$p(\mathrm{y}|\mathrm{x}, \lambda) = \frac{1}{Z(x)} exp\left(\sum_m \lambda_m F_m(\mathrm{y}, \mathrm{x})\right), \quad (3)$$

where Z(x) is the partition function or normalization factor, x is the observed input sequence which is a vector of vectors and y is the output label sequence. CRFs enable us to exploit a rich collection of interdependent features observed in the input sequence. During the training phase, the parameters of the model λ_m and μ_s are determined.

Inference is used to calculate the most likely sequence y given a new input x. Algorithms for

dynamic programming, such as the Viterbi algorithm, can be used.

4 Implementation and Results Analysis

This section describes the system's implementation and summarises the results of a POS tagging experiment conducted on a custom-built corpus of 30647 words.

4.1 Software Environment

The experiment is performed in a Python Anaconda distribution as well as in a cloud-based Google colab environment. Python libraries such as NLTK, Sklearn CRFsuite-0.3.6, Matplotlib, and Elisa are used in testing the model as well as visualization of results.

4.2 Tagset Distribution in the Corpus

Fig. 1 gives the frequency distribution of the five most frequently used tags in the corpus. A shown in the graph, verb base form (VB) has the highest number of occurrences (4362) followed by a common noun (CMN, 3434 instances), personal pronoun (PSP, 3287 instances), proper noun (PPN, 3102 instances) and Adverb base form (RB) with 2805 instances.

4.3 Transitions and Weights Learned by the Model

The conditional Random Field model learns the transitional relationship between tags in the training corpus and assigns weights accordingly. It is a measure of the relationship that exists between output sequences. Tags with higher transition probabilities are given more weight. Fig. 2 highlights transitions between tags involved in the top 10 most frequent transitions in the training corpus.

As seen from Fig. 2, the CRF model learned that if a given word is tagged as a Double Adverb (DRB), it is likely to be followed by a Double Adverb(DRB).

Fig. 3 contains a list of the top 20 most unlikely transitions found in the training corpus.

Table 2. The Mizo tagged corpus summary

Particulars	Count
Total no. of words	30647
Total no. of sentences	968
Total no. of unique tags	41
Total no. of unique vocabulary	4885
Most frequent word	a (2602 times)
Most frequent tag	VB (4632 times)

Table 3. Potential features

Name of features	Selected contents
Word	Current token under consideration
postag-1	Previous token POS tag
postag+1	Next token POS tag
is_first	First token in a sentence
is_last	Last token in a sentence
is_capitalized	First character is capitalized
is_all_caps	All characters are capitalized
is_all_lower	All characters are in lowercase
prefix-1	First character of a token
prefix-2	First two characters of a token
prefix-3	First three characters of a token
suffix-1	Last character of a token
suffix-2	Last two characters of a token
suffix-3	Last characters of a token
prev_word	Previous token
next_word	Next token
has_hyphen	Whether a token contains a hyphen
is_numeric	Whether a token consists of numbers only
capitals_inside	Capital letter other than first character

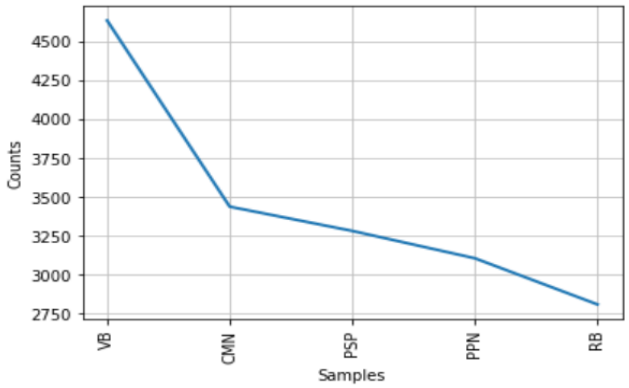


Fig. 1. Frequency distribution of five most frequently used tags in the corpus

From \ To	DRB	DJJ	ET	VB	IP	JJ	NG	SJJ	PT	CJJ	PT	;	PPN	AT	MP
DRB	3.743	0.0	0.0	-0.805	0.0	-0.421	0.175	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
DJJ	0.0	3.219	0.0	0.0	0.0	0.115	0.0	0.0	0.0	0.0	0.0	0.0	-0.583	0.0	0.832
ET	0.0	0.0	2.682	0.0	0.0	-0.219	0.0	0.0	0.0	0.0	0.0	0.0	-0.811	0.0	0.0
VB	0.986	-0.555	-0.031	-0.514	-0.729	-0.183	2.291	-0.669	0.718	-0.798	0.718	0.0	0.168	0.012	-0.269
IP	0.0	0.0	0.0	-0.146	0.0	0.0	2.168	0.0	0.606	0.0	0.606	0.0	-0.007	0.0	0.0
JJ	1.022	-0.302	0.0	-0.214	0.233	0.098	1.283	1.829	-0.11	2.207	-0.11	0.0	0.341	0.402	0.0
NG	0.0	0.0	0.379	-1.542	0.0	0.199	0.0	0.0	0.683	0.003	0.683	0.0	-1.827	0.194	-0.033
SJJ	0.0	0.0	0.0	0.52	0.0	-0.363	0.0	0.0	-0.553	0.0	-0.553	0.0	0.0	0.4	0.0
PT	0.0	0.0	0.0	-0.404	0.0	-0.27	-0.083	0.0	1.223	0.0	1.223	2.015	-0.718	0.0	0.0
CJJ	0.0	0.0	0.0	-0.144	0.0	-0.082	0.0	0.0	0.474	0.0	0.474	0.0	0.0	0.0	0.0
PT	0.0	0.0	0.0	-0.404	0.0	-0.27	-0.083	0.0	1.223	0.0	1.223	2.015	-0.718	0.0	0.0
;	0.0	0.0	0.0	0.0	0.0	-0.399	0.0	0.0	0.0	0.0	0.0	0.0	0.276	0.0	0.595
PPN	0.0	-0.893	-0.211	0.404	0.0	0.429	-0.986	0.0	-0.381	0.0	-0.381	0.0	1.421	1.541	0.0
AT	0.0	0.0	0.329	-0.07	0.0	0.061	-2.728	0.0	-0.503	0.0	-0.503	1.379	0.476	-0.478	-0.24
MP	0.0	0.594	0.0	0.0	0.0	-0.418	0.0	0.0	0.388	0.0	0.388	0.0	0.026	-1.882	1.903

Fig. 2. Transition weights between tags of top 10 likely transitions. (Indicated by dark green cells)

Fig. 3 shows that from the training set, transitions from Article (AT) to a Negation (NG) is highly unlikely. Negative weights represent impossible transitions in the training corpus.

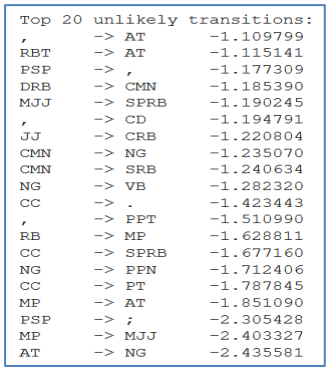
4.4 Feature Based on Context of a Given Word Selected by the Model

The CRF model learns each word's context in the corpus through training and assigns the calculated weight to each feature for each tag.

Fig. 4 highlights the top features selected for the feature of tags such as Abstract Noun (ABN), Article (AT), Co-ordinating conjunction (CC) and Cardinal Number (CD).

The model employs 8806 attributes, 741 transition features, and 15139 state features in total. As seen from the above tables, the features selected by the CRF model and weights assigned to them are fairly accurate representation for categorizing a given word to a probable tag.

It also demonstrates that the context of a word is crucial in determining the tag of that word. For instance, consider a feature selected for Abstract Noun (ABN). The feature 'suffix-2: na' (The last two characters of a word is 'na') is given a high weight value of 5.609. This is an accurate selection since



```
Top 20 unlikely transitions:
,       -> AT       -1.109799
RBT     -> AT       -1.115141
PSP     -> ,        -1.177309
DRB     -> CMN      -1.185390
MJJ     -> SPRB     -1.190245
,       -> CD       -1.194791
JJ      -> CRB      -1.220804
CMN     -> NG       -1.235070
CMN     -> SRB      -1.240634
NG      -> VB       -1.282320
CC      -> .        -1.423443
,       -> PPT      -1.510990
RB      -> MP       -1.628811
CC      -> SPRB     -1.677160
NG      -> PPN      -1.712406
CC      -> PT       -1.787845
MP      -> AT       -1.851090
PSP     -> ;        -2.305428
MP      -> MJJ      -2.403327
AT      -> NG       -2.435581
```

Fig. 3. The top 20 most unlikely transitions in the corpus

y=ABN top features		y=AT top features		y=CC top features		y=CD top features	
Weight²	Feature	Weight²	Feature	Weight²	Feature	Weight²	Feature
+5.609	suffix-2:na	+2.414	prefix-3:chu	+3.537	word:leh	+4.116	is_numeric
+3.293	suffix-3:kna	+2.290	prev_word:ko	+3.232	prefix-2:ti	+3.225	is_all_caps
+3.150	suffix-3:nna	+2.273	word:chuan	+3.040	suffix-3:uan	+2.280	prev_word:nuai
+2.740	suffix-3:hna	+2.211	suffix-3:uan	+3.026	prev_word:lamte	+2.110	suffix-2:li
+2.540	next_word:chhung	+2.148	word:chu	+3.006	word:chuan	+2.097	suffix-3:nih
+2.526	prev_word:MPCPSR	+1.974	suffix-3:chu	+2.774	prefix-3:ava	+2.054	suffix-3:hum
+1.968	prefix-3:mam	+1.904	suffix-2:hu	+2.774	prefix-2:av	+1.959	next_word:%
+1.966	word:mamawhte	+1.872	prev_word:lam	+2.741	suffix-3:hse	+1.897	word:thum
+1.960	suffix-3:mna	+1.809	postag-1:PPN	+2.542	prev_word:tawng	+1.847	word:khat
… 319 more positive …		+1.643	prev_word:lai	+2.536	suffix-3:tih	*… 246 more positive …*	
… 86 more negative …		*… 114 more positive …*		*… 382 more positive …*		*… 53 more negative …*	
-3.193	prefix-2:hn	*… 31 more negative …*		*… 64 more negative …*		-2.073	postag-1:ET

Fig. 4. Weight of different features for ABN, AT, CC, and CD

Table 4. Performance of the tagger

Train set: Test set	Accuracy	F1-score	Precision	Recall
70:30	89.16%	89.05%	89.11%	89.16%
75:25	89.41%	89.30%	89.34%	89.41%
80:20	89.05%	88.91%	88.95%	89.05%
85:15	89.87%	89.47%	89.78%	89.87%
90:10	89.81%	89.77%	89.92%	89.82%
Average Score	**89.46%**	**89.3%**	**89.42%**	**89.48%**

most words ending with ‘na’ tend to be an Abstract Noun in Mizo, e.g., Hmangaihna, duhsakna, thiamna, etc. Similarly, the ‘is_numeric’ feature is given a large weight value since any numeric value is likely to represent a Cardinal Number.

4.5 Performance Evaluation

The corpus is split into a training dataset and test dataset to assess the proposed CRF-based tagger's performance. Results are observed for various split ratios such as 70:30, 75:25, 80:20, 85:15, and 90:10 for train and test set, respectively.

The tagger's performance is assessed using a variety of metrics such as accuracy, precision, recall, and f1-score, with the results shown in table 4. The system yielded an average score of 89.46% accuracy, 89.3 % F1-score, 89.42 % precision, and 89.8% recall. It can be observed that the accuracy of the CRF model tagger appears to improve as the

size of the tagger corpus grows. From the corpus size of 15000 onwards, only a slight increase in accuracy is observed for each addition of corpus text.

This indicates that a higher accuracy can be obtained from a larger corpus. Features selected in Table 3 are considered fairly sufficient since adding more context features does not show much improvement in the result obtained.

5 Conclusion and Future Works

The proposed model provided in this research work serves a ground work for further research for Mizo language in the field of NLP. A tagged corpus of 30647 words is created which is a significant addition to the low resource language. Suitability of stochastic based tagger for Mizo language is checked by using Conditional Random Field model. Results showed that it provides a fairly good representation of the language. Our future work consists of creating larger tagged corpus and testing the suitability of other models for the language.

Acknowledgments

The authors would like to thank Mizoram University and National Institute of Technology Silchar for supporting this research work.

References

1. **Awasthi, P., Rao, D., Ravindran, B. (2006).** Part of speech tagging and chunking with HMM and CFR. Proceedings of NLP Association of India (NLPAI) Machine Learning Contest.
2. **Deshmukh, R.D., Kiwelekar, A. (2020)**. Deep learning techniques for part of speech tagging by natural language processing. 2nd International Conference on Innovative Mechanisms for Industry Applications IEEE. (ICIMIA), pp. 76–81 DOI: 10.1109/ICIMIA48430.2020.9074941.
3. **Singh, T.D., Ekbal, A., Bandyopadhyay, S. (2008).** Manipuri POS tagging using CRF and SVM: A language independent approach. Proceeding of 6th International Conference on Natural Language Processing (ICON-2008), pp. 240–245.
4. **Pandian, S.L., Geetha, T.V. (2009).** CRF models for Tamil part of speech tagging and chunking. International Conference on Computer Processing of Oriental Languages, pp. 11–22. Springer. DOI: 10.1007/978-3-642-00831-3_2.
5. **Outahajala, M., Benajiba, Y., Rosso, P., Zenkouar, L. (2011).** Pos tagging in Amazighe using support vector machines and conditional random fields. International Conference on Application of Natural Language to Information Systems, Springer, pp. 238–324. DOI: 10.1007/978-3-642-22327-3_28.
6. **Nongmeikapam, K., Bandyopadhyay, S. (2012).** A transliteration of CRF based Manipuri POS tagging. Procedia Technology, Vol. 6, pp. 582–589. DOI: 10.1016/j.protcy.2012.10.070.
7. **Shambhavi, S., Kumar, R. (2012).** Kannada part-of-speech tagging with probabilistic classifiers. International Journal of Computer Applications, Vol. 48, No. 17, pp. 26–30. DOI:10.5120/7442-0452.
8. **Ojha, A.K., Behera, P., Singh, S., Jha, G.N. (2015).** Training & evaluation of POS taggers in Indo-Aryan languages: A case of Hindi, Odia and Bhojpuri. 7th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, pp. 524–529.
9. **Ghosh, S., Ghosh, S., Das, D. (2016).** Part-of-speech tagging of code-mixed social media text. The second workshop on computational approaches to code switching, pp. 90–97.
10. **Zeroual, I., Lakhouaja, A., Belahbib, R. (2017).** Towards a standard part of speech tagset for the Arabic language. Journal of King Saud University-Computer and Information Sciences, Vol. 29, No. 2, pp. 171–178. DOI: 10.1016/j.jksuci.2017.01.006.
11. **Dibitso, M.A., Owolawi, P.A., Ojo, S.O. (2019).** Part of speech tagging for Setswana African language. International Multidisciplinary Information Technology and Engineering Conference (IMITEC) IEEE, pp. 1–6.
12. **Lafferty, J., McCallum, A., Pereira, F.C. (2001).** Conditional random fields: Probabilistic models for segmenting and labeling sequence data. https://repository.upenn.edu/cis_papers/159/.
13. **Lalzarzova, K. (2016)**. Mizo Tawng Grammar & Composition. K. Sangzawna, Aizawl, Mizoram.
14. **Thangzikpuia,P.C. (2019).** Mizo Tawng Grammar.
15. **Lalhluna, R.K, (2014)**. Cinque Foils – Zo Ṭawng Grammar.
16. **Mizoram Board of School Education (2020)**. Mizo Ṭawng Ziah dan.

17. **Khiangte, Laltluangliana (1997).** Thuhlaril Aizawl: College Text Book. Editorial Board Publications.
18. **Pakray, P., Pal, A., Majumder, G., Gelbukh, A. (2015)**. Resource building and parts-of-speech (POS) tagging for the Mizo language. 4th Mexican International Conference on Artificial Intelligence (MICAI). pp. 3–7.
19. **Nunsanga, M.V., Pakray, P., Lalngaihtuaha, M., Singh, L.L.K. (2021).** Part-of-speech tagging in Mizo language: A preliminary study. Data Intelligence and Cognitive Informatics, pp. 625–635, Springer. DOI: 10.1007/978-981-15-8530-2_49.

Article received on 12/06/2021; accepted on 04/9/2021. Corresponding author is Partha Pakray.