CITA2023

Neural Sequence Labeling based Sentence Segmentation for Myanmar Language

Ye Kyaw Thu, Thura Aung and Thepchai Supnithi

Language Understanding $\lambda(\mu)$ Laboratory

NECTEC
a member of NSTDA

Table of Content

- Introduction
- Related Works
- Corpus Development
- Methodology (NCRF^{++})
- Experimental Setup
- Results and Discussion
- Conclusion

Introduction

- The task of segmenting text into sentences that are independent units and grammatically linked words
- In the formal Myanmar language, sentences are grammatically correct and typically end with a "။" pote-ma.
- Informal language is more frequently used in daily conversations with others due to its easy flow.
- There are no predefined rules to identify the ending of sentences in informal usages for the machine itself.
- Some of the applications based on conversations, e.g, Automatic Speech Recognition (ASR), Speech Synthesis or Text-to-Speech (TTS), and chatbots, need to identify the end of sentences.

Related Works

- Win Pa Pa, Ye Kyaw Thu, Finch,A. , Sumita, E.: Word Boundary Identification for Myanmar Text Using Conditional Random Fields, In Zin, T., Lin, JW., Pan, JS., Tin, P., Yokota, M., (eds) Genetic and Evolutionary Computing. GEC 2015. Advances in Intelligent Systems and Computing, vol 388. Springer, Cham (2016).
- Ye Kyaw Thu, Finch, A., Sagisaka, Y., Sumita, E.: A Study of Myanmar Word Segmentation Schemes for Statistical Machine Translation, In Proceedings of the 11th International Conference on Computer Applications, pp. 167-179, Yangon, Myanmar (2013).

Related Works

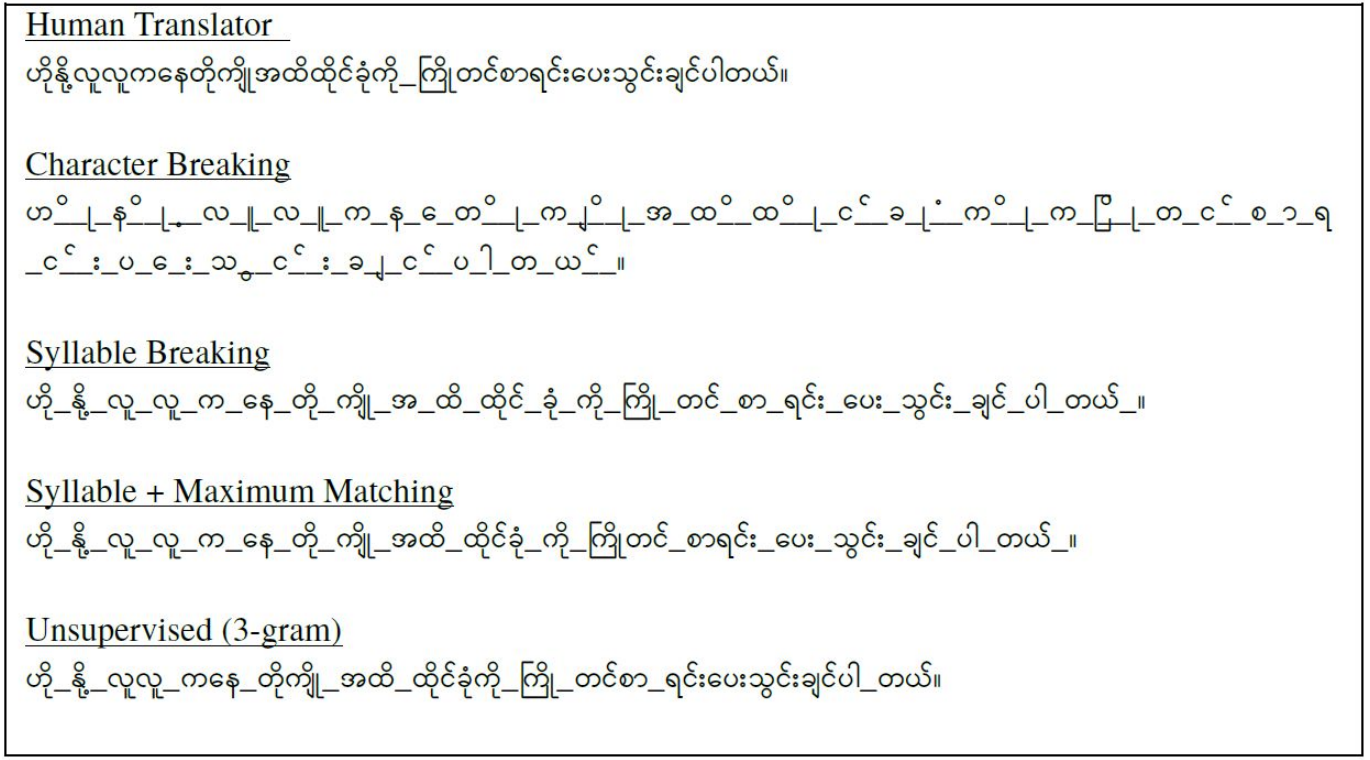

Fig 1. Different segmentation methods for a Myanmar sentence (Ye Kyaw Thu et al., 2013)

Related Works

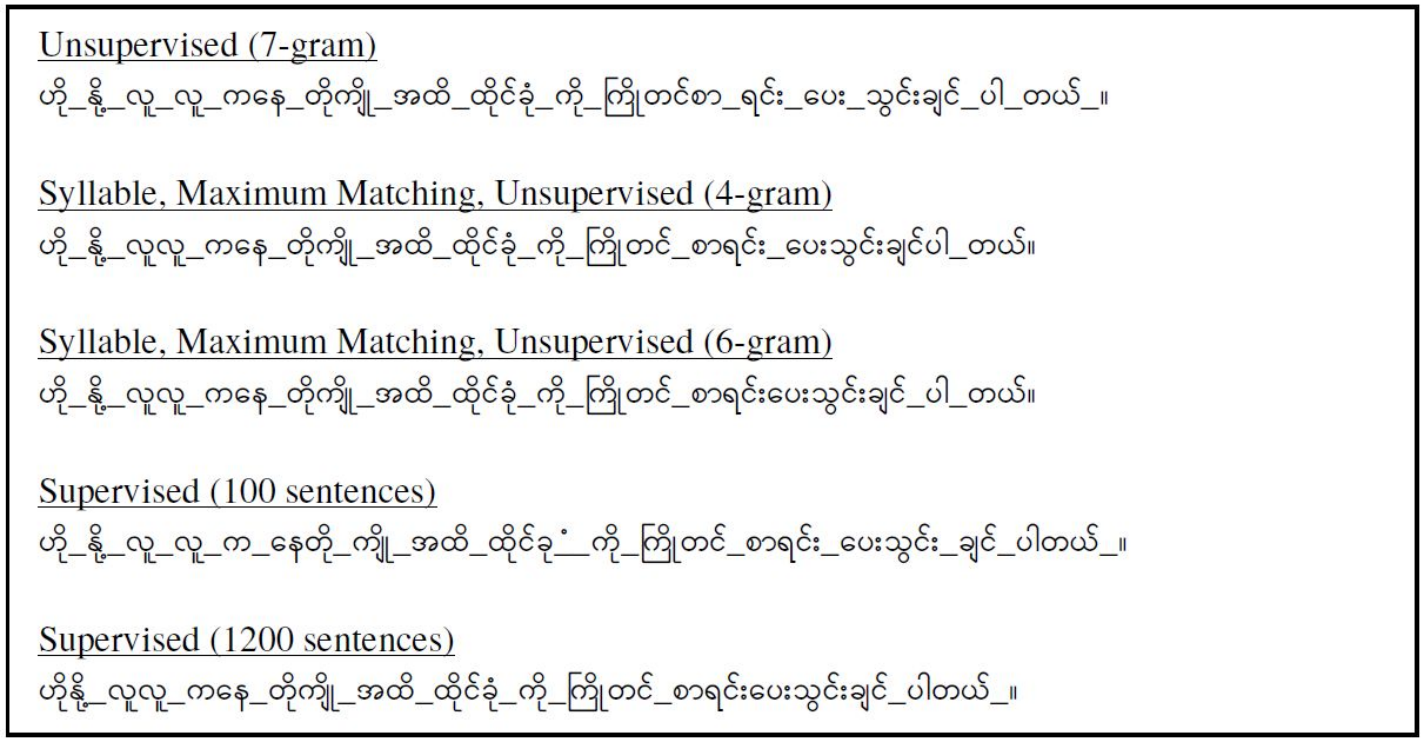

Fig 2. Different segmentation methods for a Myanmar sentence (Ye Kyaw Thu et al., 2013)

Related Works

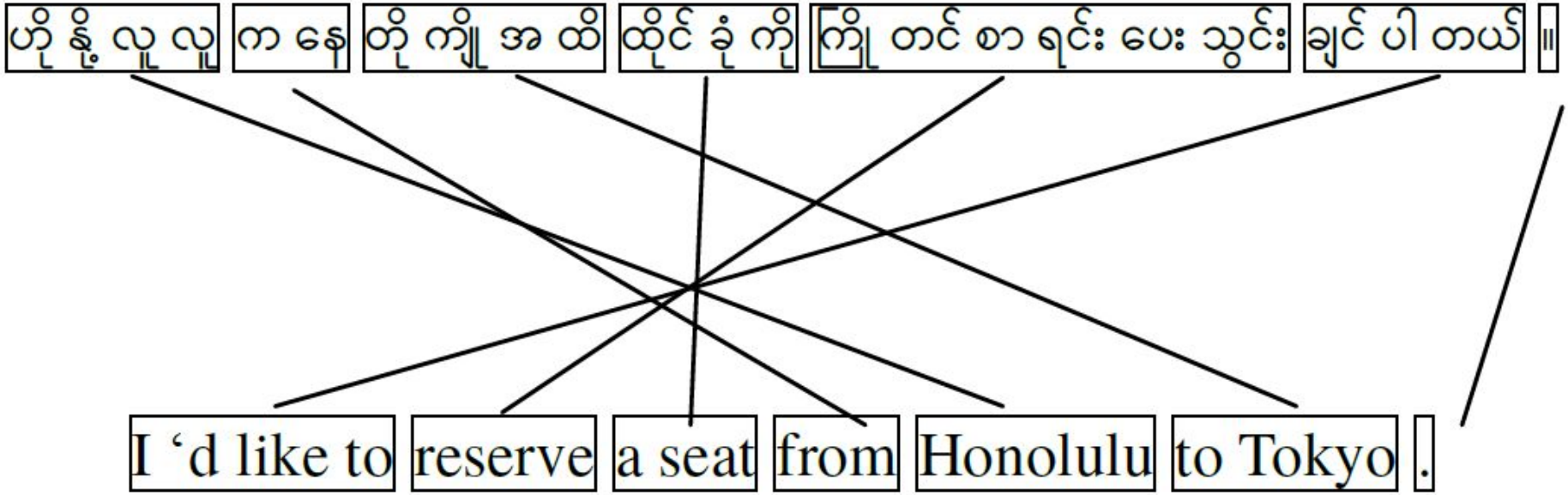

Fig 3. A syllable-to-word aligned Myanmar-English sentence pair (Ye Kyaw Thu et al., 2013)

Related Works

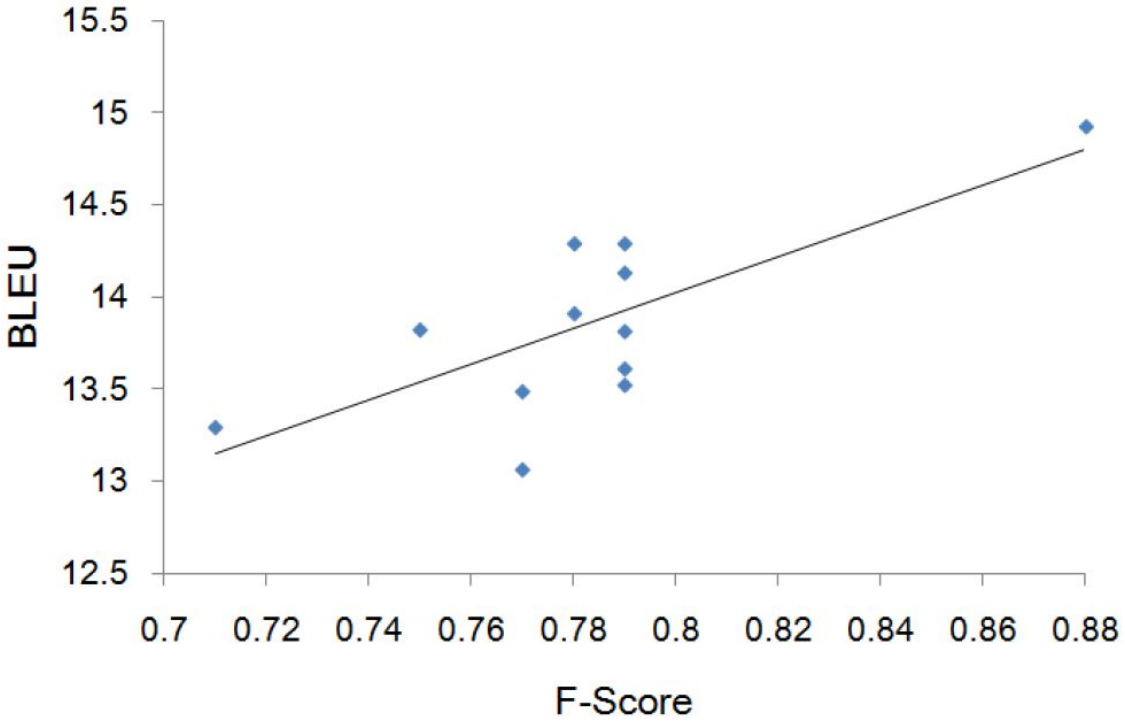

Fig 4. The correlation between BLEU and segmentation F-score for my-en (Ye Kyaw Thu et al., 2013)

Related Works

- Sadvilkar, N., Neumann, M.: PySBD: Pragmatic Sentence Boundary Disambiguation, In Proceedings of Second Workshop for NLP Open Source Software (NLPOSS), Association for Computational Linguistics, pp. 110-114. (2020)
- Authors introduced a multilingual rule-based sentence segmentation tool called PySBD in which Myanmar sentence segmentation is available but it is only useful for formal usages because sentence segmentation is based on the sentence delimiter "။" pote-ma, which is not used in informal communications.

Corpus Development (mySentence)

- Myanmar NLP researchers are facing many difficulties arising from the lack of resources; in particular parallel corpora are scarce.
- For this reason, we annotated text data manually with mySentence tag information.
- The myPOS corpus version 3.0 consists of 43,196 meaningful word sequences written in formal and informal formats from various domain areas and the whole corpus has already been word-segmented manually.
- We also collected Myanmar sentences and paragraphs from different online resources such as Facebook and Wikipedia and from the short stories available on Facebook pages.

Corpus Development (mySentence)

Data Resources	sentences	paragraphs
myPOS version 3.0	40,191	2,917
Covid QandA	1,000	1,350
Facebook posts	93	672
Wikipedia articles	2,780	1,060
Nay Win Myint's short stories	220	1,150
Nikoye's short stories	327	735
Maung Zi's myth stories	2,516	581
Total	**47,127**	**8,465**

Table 1. Statistics of sentence segmentation dataset

Corpus Development (mySentence)

Tag	Frequency	Proportion
B	47,264	7.24%
E	48,690	7.33%
N	137,592	20.46%
O	436,942	64.97%

Table 2. Statistics of tags in the mySentence corpus.

Methodology (NCRF^{++})

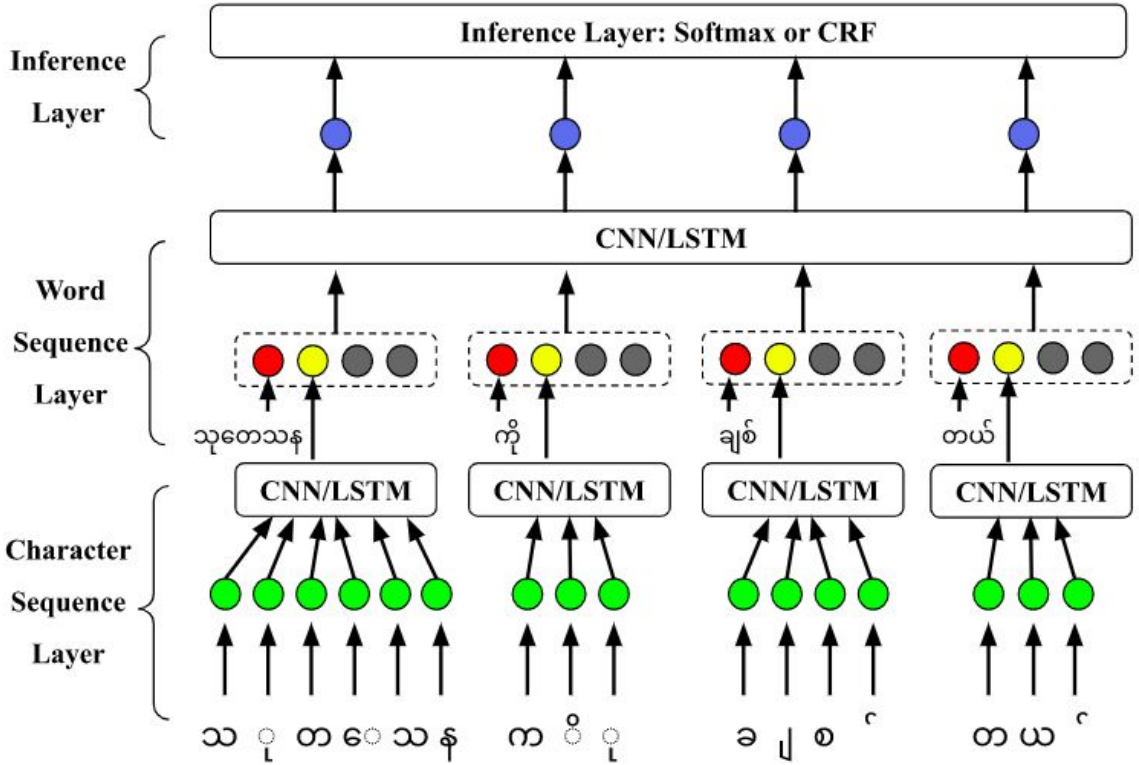

Fig 5. NCRF++ for a Myanmar language sentence example

Methodology (NCRF^{++})

- **Embedding Layer**: The input sequences (like words in a sentence) first get represented as embeddings. These can be word embeddings like Word2Vec, GloVe, or character embeddings for character-level information.
- **Encoder Layer**: After getting the embeddings, they are passed through an encoder, typically a recurrent neural network (RNN), long short-term memory (LSTM), or a gated recurrent unit (GRU). This helps capture contextual information of the sequence.
- **CRF Layer**: On top of the neural network structure, a CRF layer is applied. CRFs help in ensuring that the sequence of labels output by the model is coherent. Instead of just predicting each label independently, the CRF layer takes into account the surrounding labels to make a decision.

Experimental Setup

	sentence	**sentence + paragraph**
train	40,000	47,000
validation	2,414	3,079
test	4,712	5,512

Table 3. Dataset splitting for the experiments.

Experimental Setup

- The word distribution with Zipf's law between two datasets measured with top 1,000 words for 1-gram and 2-gram are as shown in Fig 6.
- The Zipf curves for the two datasets are almost identical and show the similarity of word distributions.

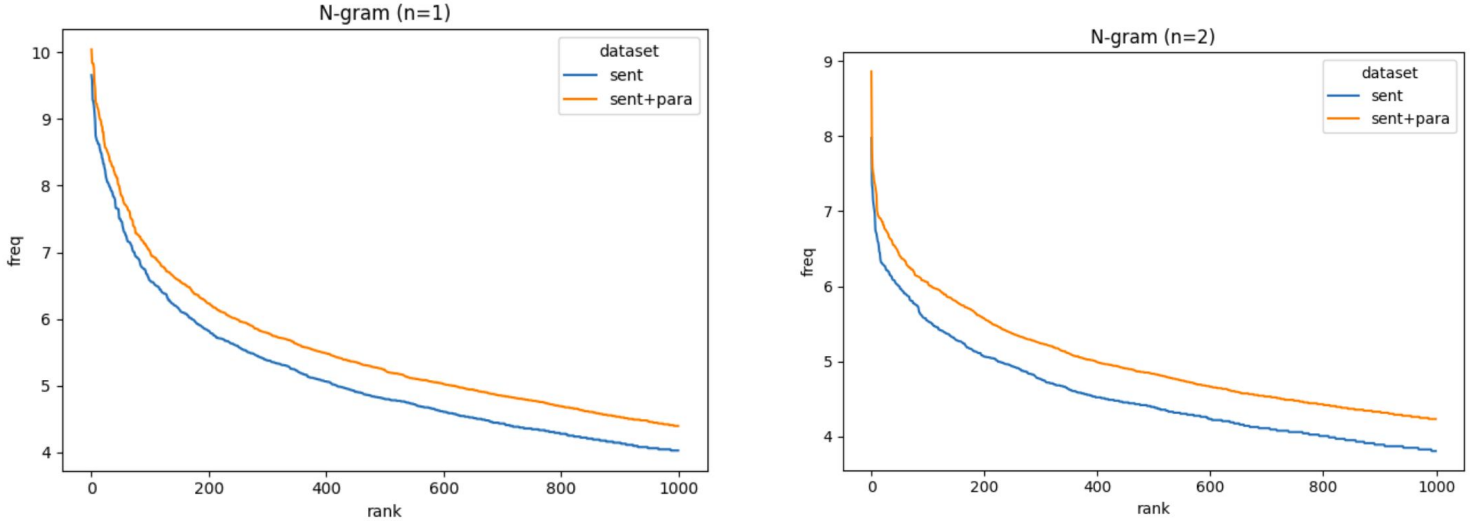

Fig 6. Zipf's law distributions in 1-gram and 2-gram analysis of word between sentence only and sentence+paragraph datasets.

Experimental Setup

Parameter	Value	Parameter	Value
char emb size	30	word emb size	50
char hidden	50	word hidden	200
CNN layer	4	CNN kernel size	3
dropout rate	0.5	batch size	10
L2 regularization λ	1e-8	learning rate decay	0.05
epochs	100	optimizer	SGD

Table 4. Hyperparameters used in experiments.

Results and Discussion

	Test Data	wCNN + Softmax	wCNN + CRF	wLSTM + Softmax	wLSTM + CRF
NoChar	sent sent + para	99.92 93.58	**99.95** **93.63**	**99.95** **93.63**	**99.95** **93.63**
cCNN	sent sent + para	**99.95** **93.63**	**99.95** **93.63**	**99.95** **93.63**	92.02 87.65
cLSTM	sent sent + para	**99.95** **93.63**	**99.95** **93.63**	**99.95** **93.63**	99.91 93.59

Table 5. Accuracy % comparison of sentence-level models

(c = Character, w = Word, sent = sentence and para = paragraph)

The bold results show the highest accuracies achieved in each test data.

Results and Discussion

	Test Data	wCNN + Softmax	wCNN + CRF	wLSTM + Softmax	wLSTM + CRF
NoChar	sent sent + para	99.41 96.82	99.49 96.25	99.44 **97.40**	86.44 96.61
cCNN	sent sent + para	99.26 96.87	99.27 96.17	74.81 74.69	86.44 83.13
cLSTM	sent sent + para	**99.66** 96.36	99.49 96.04	99.49 97.29	99.56 96.61

Table 6. Accuracy % comparison of sentence+paragraph-level models

(c = Character, w = Word, sent = sentence and para = paragraph)

The bold results show the highest accuracies achieved in each test data.

Results and Discussion

Freq	Confusion Pair (REF ⇒ HYP)
7078	N ⇒ O
1951	O ⇒ N
1229	E ⇒ O
1224	B ⇒ O
48	B ⇒ N

Table 7. The Top 5 confusion pairs of sent cCNN+wLSTM+CRF model tested on sent+para test data (87.65% accuracy)

Results and Discussion

- According to the comparison of twelve NCRF++ architectures trained and tested on both sentence and sent+para data, we can see that the word LSTM with softmax inference layer and no character representation layer had the best accuracy with sent-level (99.95%) as well as sent+para-level (97.40%) data.
- According to the error analysis, most of the errors occurred because the models falsely recognized "O" tags, which have the highest proportion in the dataset.

Conclusion

- **Feature Learning**: Unlike traditional CRFs that require manual feature engineering, NCRF^{++} can automatically learn features.
- **End-to-End**: The model can be trained in an end-to-end fashion, which simplifies the process and can lead to better results.
- **Flexibility**: NCRF++ provides flexibility to use various types of embeddings and neural architectures.
- **Performance**: The integration of deep learning and CRFs often leads to state-of-the-art results in sequence labeling tasks.

Conclusion

- In the future, we will investigate the impact of pre-trained word embeddings on this sequence labeling based sentence segmentation task with different embedding settings for a low-resource language-Myanmar.
- We make all our configuration files, code, data, and models publicly available (https://github.com/ye-kyaw-thu/mySentence).

CITA2023

Thank you!

Cảm ơn!