Developing Name Entity Recognition for Structured and Unstructured Text Formatting Dataset

Nadhia Salsabila Azzahra[1,4], Muhammad Okky Ibrohim[2], Junaedi Fahmi[3], Bagus Fajar Apriyanto[3], Oskar Riandi[3]
[1]*Faculty of Informatics, Telkom University, Indonesia*
[2]*Faculty of Computer Science, Universitas Indonesia, Indonesia*
[3]*PT. Bahasa Kinerja Utama, Indonesia*
[4]*Research Internship Student, PT. Bahasa Kinerja Utama, Indonesia*
nadhiasalsabila@student.telkomuniversity.ac.id, okkyibrohim@cs.ui.ac.id, {jun, bagus, oskar}@bahasakita.co.id

***Abstract*—Named-Entity Recognition (NER) is a task that extracts the entity information from dataset into several different entity classes. Most of current NER research train the NER model from structured data such as news and Wikipedia article. Whereas, there are several tasks that generate an unstructured dataset such as speech-to-text task. In this paper, we did NER research for unstructured text formatting dataset in Indonesian language using deep learning approaches including LSTM, Bidirectional LSTM (Bi-LSTM), GRU, Bidirectional GRU (Bi-GRU), and Convolutional Neural Network (CNN). We used NERGRIT CORPUS as our dataset and modified the dataset into four types of structured and unstructured datasets. Afterward, we run several experiments scenario by combining all types of data modification and the deep learning algorithms that we used and we obtain that the highest $F1-Score$ was obtained when using Bi-GRU for standard dataset, lowercase with punctuation dataset, lowercase without punctuation dataset, and lowercase and clean dataset equal to 71.04%, 70.61%, 68.12%, and 67.45%, respectively.**

***Index Terms*—name entity recognition, unstructured text formatting, deep learning**

I. Introduction

Named-Entity Recognition (NER) is one of the important sub-task in Natural Language Processing (NLP) research. NER is a necessary step for question answering [1], text summarization [2], information extraction task [3] and to improve the quality of machine translation [4]. NER manages to identify proper name entities such as person, organization, location, event, values, etc. Several techniques have been implemented for NER that depend on manual features and pays attention to improve feature selection such as ruled-based approaches [1], supervised-learning approaches [5], unsupervised learning approches [6], etc.

NER has been frequently studied in structured format text. For example, the used of LSTM-CRF model on CoNLL-2002 [5] and CoNLL2003 [7] datasets with several languages in which models rely on two sources of information about words, which are character-based word representations learned from the supervised corpus and unsupervised word representations learned from unannotated corpora [8]. The used of hybrid Bi-LSTM and Convolutional Neural Network (CNN) architecture and also propose a novel method of encoding partial lexicon matches in neural networks and compare it to existing approaches that competitive on CoNLL2003 [7] and OntoNotes datasets [9], and the used of ensemble supervised learning algorithm including simple logistic, direct scheme, and features combination of word-level, sentence-level, and the lookup list for Indonesia Named-Entity Recognition on newspaper article [10].

There are several studies in NER that used deep learning approaches including Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM), Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU), and Convolutional Neural Network (CNN). In [11] they present NER system for Bangla online newspaper dataset that not publicly available. They used GRU as the model to identify four named entities including Person, Organization, Location, and Day. They obtain $F1-Score$ equal to 98% on training set and 69% on test set. In 2018, [12] presents NER using Bi-LSTM as their model. By using CONLL2003 [7] dataset they obtain 91.48% of $F1-Score$. Furthermore, in [13] they present LSTM model on several tasks including NER. By using LSTM on CONLL2003 [7] they obtain 90.5% of $F1-Score$. CNN model was used in [14] for Chinese NER. They used CNN model on several dataset such as OntoNotes [15], MSRA [16], Weibo NER [17], and Resume NER [18] for each datasets they obtain 74.45%, 93.71%, 59.92%, and 95.11% of $F1-Score$, respectively. Lastly, in [19] they present Arabic NER using Bi-GRU model on ANERcorp [20] dataset. By using this model they obtain $F1-Score$ equal to 89.74%.

However, there are several problems that can not use the structured text formatting dataset, such as analysing text of speech-to-text results that do not have a good structured text formatting and analyzing the unstructured text in tweets. There are several studies that used unstructured format text on tweets for NER research. In 2016, [21] presented NER for twitter using a hybrid model of Bi-LSTM and CNN on dataset that was provided by W-NUT 2016[1]. Next, in 2018, [22] did research on NER for Indonesian tweets using Conditional Random Field (CRF) classification on formal tweets, informal

[1] http://noisy-text.github.io/2016/index.html

TABLE I
NAMED-ENTITY CLASS USED IN STUDY

Class	Class Abbreviation	Example
Cardinal	CRD	*5,68 juta* (5,68 million), *dua* (two), *26,58 juta jiwa* (26,58 million people)
Date	DAT	*Senin* 4/8/2018 (Monday 4/8/2018), 7-24 *Juni* 2018 (7-24 June 2018), April 2018
Event	EVT	*Operasi Ketupat* (Ketupat Operation), *Idul Fitri* (Eid Al-Fitr)
Facility	FAC	*Sekolah Lanjutan Pertama* (Junior High School), *tol* Jakarta - Cikampek (Jakarta - Cikampel Highway)
Geopolitical	GPE	*Jakarta Timur* (East Jakarta), Indonesia, *Amerika Serikat* (United State of America), *Kabupaten* Simalungun (Simalungun District)
Language	LAN	*Bahasa Rusia* (Russian language), *Bahasa Indonesia* (Indonesia language), Latin
Law	LAW	*Hak Kekayaan Intelektua*l (intellectual Property Rights), *pasal 13 UU Ayat 2* (Article 13 section 2 of the constitution),
Location	LOC	*Taman Nasiona*l Moyo Satonda (National Park of Moyo Satonda), *Pulau* Sumbawa (Sumbawa island), *Perairan* Sumbawa (Sumbawa waters)
Money	MON	Rp 3.000 (Three thousand Rupiah), AUS$33, 66, US$ 70
Ordinal	ORD	*Tiga* (Three), 3, *Keenam kalinya* (Sixth time), *Pertama* (First)
Organization	ORG	PT Kereta Commuter Indonesia (Kereta Commuter Indonesia Inc.), World Bank
Person	PER	Andika Putra, Anjar Nugroho, Aditya
Percentage	PRC	*50 persen* (50 percent), 10%
Product	PRD	Life Jacket, Figura Shisha Lion, Chicken Grill
Quantity	QTY	19,162 gram, *Dua jam* (Two hours), 19 tahun (19 years), *Satu sampai tiga bulan* (One until three months)
Religion	REG	Islam, Budha (Buddha), Kristiani (Christian)
Time	TIM	17.00 WIB (17.00 PM) , 08.00 - 15.00 WIB (08.00 AM - 15.00 PM), Pukul 07.30 (07.30 AM)
WoArt	WOA	Layang-layang Bali (Bali kites), Lambang Garuda Pancasila (Garuda Pancasila emblem), Bhinneka Tunggal Ika (Unity in Diversity)
Political Organization	NOR	*Kepala BNN RI* (Chief of BNN of Indonesia Republic), *Kepala Dinas Pariwisata* NTB (Head of Tourism Authorities of NTB)

tweets, and combined tweets. Although tweet and the results of speech-to-text do not have a good structured text formatting, yet the structure format in twitter and speech-to-text are not similar. In twitter the *out-of-vocabulary* (OOV) appear more frequently due to human writing style such as abbreviate words, typographical error, etc. Whereas, the results in speech-to-text merely dealing with lowercase and no punctuation. Therefore, in this research, we modified the dataset from NERGRIT CORPUS [23] into four different types of datasets including standard dataset, lowercase with punctuation dataset, lowercase without punctuation dataset, and lowercase and clean dataset. These types of modifications were done to address several unstructured text formatting datasets like speech-to-text output that we explain before.

In this paper, we used several deep learning approaches such as LSTM, Bi-LSTM, GRU, Bi-GRU, and CNN. Since those algorithms gave a good performance on previous NER researches in [8], [24], and [10].

The rest of paper is organized as follows. Section II describes briefly of our related works. Section III describes the dataset and methodology we used in this research. Furthermore, Section IV presents experimental results on datasets using several models. Lastly, Section V describes our conclusions and future work suggestion from this work.

II. RELATED WORKS

The used of deep learning approach for NER research has shown the good performance for English language dataset [9]. Unfortunately, for Indonesian language is still rarely done because of the availability of dataset. In 2016, [9]. It uses a hybrid Bi-LSTM and CNN for NER research. They used two different dataset i.e. CONLL-2003 [7] and OntoNotes [25]. For each dataset they obtain 91.62% $F1-Score$ on CoNLL-2003 [7] and 86.28% on OntoNotes [25].

There are a few relative research on NER in Indonesian language. In [26] they presented NER for Indonesian Language using various deep approaches as hybrid models of Bi-LSTM and CNN. Extracted information from articles with the structured format text into four different classes such as Person, Organization, Location, and Event. They obtain 79.43% of $F1 - Score$ by using the hybrid models of Bi-LSTM and CNN.

In 2017, [27] presented a recurrent neural network (RNN) framework based on word embeddings and character representation for Biomedical Named-Entity Recognition. On the top of the layer they add CRF layer to jointly decode labels for the whole sentence. For the features of their model they used word embeddings and character embeddings. By using the hybrid of Bi-LSTM and CNN architecture and the features they obtain 86.55% F1 on BioCreative II gene mention (GM) corpus [28] and 73.79% F1 on JNLPBA 2004 corpus [29].

III. Dataset and Methodology

A. Dataset

The dataset that we used for training and testing data is obtained from NERGRIT CORPUS [23]. NERGRIT CORPUS is a corpus that conducted from news articles and annotated for the NER task. The output tags are annotated with ”BIO” format for indicating the position of the token in entity, which stands for *(B)eginning*, *(I)nside*, and *(O)utside* [30]. For instance, if we have the sentence ”Susi Pudjiastuti *pergi ke Makassar kemarin*.” (Susi Pudjiastuti went to Makassar yesterday.), we should BIO tag its sentence as ”*Susi*-B *Pudjiastuti*-I *pergi*-O *ke*-O *Makassar*-B *kemarin*-O .-O”. The NERGRIT dataset contained 435,344 tokens with 24,314 unique tokens, splitted into train set (309,095 tokens with 20,263 unique tokens), validation set (61,682 tokens with 7,812 unique tokens), and test set (64,567 tokens with 7,980 unique tokens). Unfortunately, the NERGRIT dataset does not have a sentence separator. In this dataset, a punctuation that sticks to word will count as a different token.

The list of named-entity class and example is described in Table I. We modified the dataset from structured text format into four types of dataset including *standard, lowercase with punctuation, lowercase without punctuation,* and *lowercase and clean dataset.* Before we modified the NERGRIT into four types dataset, we made did a data preprocessing from the NERGRIT dataset including adding a sentence separator, correcting typos and wrong labels, removing duplicate sentences, and removing unnecessary phrase/sentence (like figure captions) and character (like extra space).

1) **Standard dataset.** This dataset type is the well formatted dataset, i.e. have proper capitalization and punctuation. This dataset was built by combining the train, validation, and test NERGRIT dataset that we preprocessed as we describe before like adding sentence separator, correcting typos and wrong labels, etc. This dataset type is prepared for tagging NER in well structured data like news article. ”*Ekspor dilakukan melalui Pelabuhan Tanjung Priok, Jakarta Utara.*” (Export was done through Tanjung Priok Port, North Jakarta.) is the example of a sentence in this type of dataset and shown in Table II.

TABLE II
THE EXAMPLE OF STANDARD DATASET SENTENCE

Token	Label
Ekspor	O
dilakukan	O
melalui	O
Pelabuhan	B-FAC
Tanjung	I-FAC
Priok	I-FAC
,	O
Jakarta	B-GPE
Utara	I-GPE
.	O

2) **Lowercase with punctuation dataset.** In this type of dataset, we modified all sentences in *standard dataset* into lowercase. However, we still keep the punctuation. This dataset type was prepared for tagging NER for text that came from speech-to-text output that has punctuation. ”*ekspor dilakukan melalui pelabuhan tanjung priok, jakarta utara.*” (export was done through tanjung priok port, north jakarta.) is the example of a sentence in this type of dataset and shown in Table III.

TABLE III
THE EXAMPLE OF LOWERCASE WITH PUNCTUATION DATASET SENTENCE

Token	Label
ekspor	O
dilakukan	O
melalui	O
pelabuhan	B-FAC
tanjung	I-FAC
priok	I-FAC
,	O
jakarta	B-GPE
utara	I-GPE
.	O

3) **Lowercase without punctuation dataset.** In this type of dataset, we remove the punctuation tokens and its tag from *lowercase with punctuation dataset* in order to prepare NER tagger for lowercase text that does not have punctuation. ”*ekspor dilakukan melalui pelabuhan tanjung priok jakarta utara*” (export was done through tanjung priok port north jakarta) is the example of a sentence in this type of dataset and shown in Table IV.

TABLE IV
THE EXAMPLE OF LOWERCASE WITHOUT PUNCTUATION DATASET SENTENCE

Token	Label
ekspor	O
dilakukan	O
melalui	O
pelabuhan	B-FAC
tanjung	I-FAC
priok	I-FAC
jakarta	B-GPE
utara	I-GPE

4) **Lowercase and clean dataset** In this type of dataset, we remove the sentence and its tags for each sentence in *lowercase without punctuation dataset* that only have *O* tags for all token in order to check whether we will have better NER model or not for lowercase without punctuation dataset if we reduce the *O* tags from dataset. ”*pengiriman ini besar sekali dan dilakukan dengan sangat efisien*” (the shipping is done in a large scale and efficiently) is the example of a sentence that we removed from this type of dataset and shown in Table V.

B. Methodology

In this experiment, we divide each type of our dataset into three subsets which are training set, validation set, and test set.

TABLE V
THE EXAMPLE OF SENTENCE THAT WILL BE REMOVED FROM LOWERCASE AND CLEAN DATASET

Token	Label
pengiriman	O
ini	O
besar	O
sekali	O
dan	O
dilakukan	O
dengan	O
sangat	O
efisien	O

The proportion of the subsets is 60% for training set, 20% for validation set, and 20% for test set.

For the classifiers models, we used several deep learning approaches including LSTM, Bi-LSTM, GRU, Bi-GRU, and CNN. There were several previous works that implemented all these models as in [11] they implement GRU as a model in NER system on Bangla online newspaper dataset that not publicly available. In [12] they implement Bi-LSTM in NER on CONLL2003 [7] dataset. Furthermore, in [13] they implement the LSTM model on several tasks including NER on CONLL2003 [7]. In 2018, the CNN model was implemented in [14]. They implement the CNN model on several datasets such as OntoNotes [15], MSRA [16], Weibo NER [17], and Resume NER [18]. In 2017, [19] implement Bi-GRU as model in Arabic NER on ANERcorp [20] dataset. We develop our models using a deep learning framework namely TensorFlow[2].

*1) **Long Short-Term Memory (LSTM):*** LSTM is one of the types of deep learning approaches that have shown extreme success in modeling sequential data and shows great performance in learning long-distance dependencies [24]. For the parameters, we use 30 epochs for training the dataset with 64 of batch size and 0.5 of dropout. Furthermore, we use softmax for the activation function and Adam [31] as the optimizer.

*2) **Bidirectional Long Short-Term Memory (Bi-LSTM):*** Bi-LSTM is a development model of LSTM. The differences between LSTM and Bi-LSTM are Bi-LSTM can work on both past and future features from the input. However, LSTM only considers the previous history. The parameter we used for this model is the same as the parameters used in the LSTM model.

*3) **Gated Recurrent Unit (GRU):*** A gated recurrent unit (GRU) was introduced by [32] to make each recurrent unit to adaptively capture dependencies of different time scales. GRU has gating units that modulate the flow of information inside the unit similar as LSTM, however, without having separate memory cells [24]. The parameter we used for this model is the same as the parameters used in the LSTM model.

*4) **Bidirectional Gated Recurrent Unit (Bi-GRU):*** Bi-GRU is a development model of GRU. The differences between GRU and Bi-GRU are Bi-GRU allows for the use of information from both previous time steps and later time steps whereas GRU only uses the information from previous time steps. The parameter we used for this model is the same as the parameters used in the LSTM model.

*5) **Convolutional Neural Networks (CNN):*** CNN used convolution in place of general matrix multiplication in their layers [33]. In our model, after the embedding layer, we add feature layers, such as convolutional 1D layer and pooling layer. The filter equal to 39 and kernel size equal to 4. In convolutional layer, we add padding to avoid the decreasing number of word embedding sequences. We also add Relu as activation of this convolutional layer. For the second feature layer, we add pooling layer to reduce the spatial size of the convolved feature. Since, we flatten the final output and feed it to a regular neural network for classification purposes. For the parameters, we use 30 epochs for training the dataset, 8 of batch size, and 0.5 of dropout. Same as the previous model we describe, we also use softmax for the activation and Adam [31] as the optimizer in our CNN model.

For the metric evaluation, we use $F1-Score$ [34] to evaluate our models' performance. In this research, we evaluate our models' performance per entity, not token. Table VII shows the prediction label and its true label. If we use the token-level evaluation, the token "Susi" is correct, and the token "Pudjiastuti" is wrong. But if we evaluate our models per entity, "Susi Pudjiastuti" is a complete named entity, so the predictions for "Susi" have to be "B-PER" and "I-PER" for "Pudjiastuti". Otherwise, this is the wrong predicted entity, even token "Susi" is predicted as "B-PER". The $F1 - Score$ is a harmonic average of the precision and recall that shown in equation 1 and equation 2. The equation of $F1 - Score$ shown in equation 3.

$$Precision = \frac{TP}{(TP + FP)} \tag{1}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{2}$$

$$F1 - Score = \frac{precision \times recall}{precision + recall} \tag{3}$$

where TP is True Positive, FP is False Positive, and FN is False Negative.

IV. EXPERIMENT RESULTS AND DISCUSSIONS

Table VIII presents the results of our comparison models using LSTM, Bi-LSTM, GRU, Bi-GRU, and CNN. From table VIII, we can see that Bi-GRU gives the best performance on *standard dataset*, *lowercase with punctuation dataset*, *lowercase without punctuation dataset*, and also *lowercase and clean dataset* with the $F1 - Score$ for each dataset type is 71.04%, 70.61%, 68.12%, 67.45% respectively.

Based on our exposure, Bi-GRU can give the best performance for all datasets types since the bidirectional run the inputs in two ways, one from past to future and one from future to past and what differs this approach from unidirectional is that in the GRU and LSTM that runs backward preserved the information from the future and using the two hidden states

[2] https://www.tensorflow.org/

TABLE VI
NUMBER OF ENTRIES FOR EACH CATEGORY FOR OVERALL DATASETS

Class	Class Abbreviation	Dataset			
		Standard	Lowercase with Punctuation	Lowercase without Punctuation	Lowercase and Clean
Cardinal	CRD	4,553	4,553	4,008	4,008
Date	DAT	7,791	7,791	6,378	6,378
Event	EVT	4,549	4,549	4,464	4,464
Facility	FAC	1,788	1,788	1,706	1,706
Geopolitical	GPE	11,250	11,250	11,204	11,204
Language	LAN	23	23	23	23
Law	LAW	2,333	2,333	2,116	2,116
Location	LOC	8,707	8,707	8,549	8,549
Money	MON	5,821	5,821	5,017	5,017
Political Organization	NOR	13,550	13,550	13,370	13,370
Ordinal	ORD	718	718	698	698
Organization	ORG	9,494	9,494	9,077	9,077
Person	PER	13,784	13,784	13,624	13,624
Percentage	PRC	3,335	3,335	2,546	2,546
Product	PRD	10,245	10,245	9,937	9,937
Quantity	QTY	5,165	5,165	4,702	4,702
Religion	REG	687	687	679	679
Time	TIM	1,506	1,506	1,103	1,103
WoArt	WOA	579	579	551	551
not in any chunk	O	295,353	295,353	240,594	209,888

TABLE VII
EXAMPLE OF NER PREDICTION

Words	True Label	Predicted Label
Susi	B-PER	B-PER
Pudjiastuti	I-PER	O
adalah	O	O
seorang	O	O
Mentri	B-NOR	B-NOR
Kelautan	I-NOR	I-NOR

combined that were able in any point in time to preserve information from both past and future. Hence, using the bidirectional model obtains higher accuracy than the other models.

From *lowercase and clean* results, we see that all models give lower performance if compared to *lowercase without punctuation* results. Based on our analysis, removing the sentence that just has an *O* tag makes our models can not learn that a sentence could just has an *O* tag. It means, we cannot be balancing the dataset to improve the performance by removing the sentence that just has an *O* tag. If we want to increase the performance by balancing the dataset, we need to add a new dataset, either by manual annotation, using text augmentation, or using a particular dataset balancing algorithm.

V. CONCLUSIONS AND FUTURE WORKS

Our experiment results show that the highest performance for all dataset types reached when using the Bi-GRU model. The $F1 - Score$ for each dataset type are 71.04% for *standard dataset*, 70.61% for *lowercase with punctuation dataset*, 68.12% for *lowercase and clean dataset*, and 67.45% for *lowercase and clean dataset*. Bi-GRU considered as the best model due to bidirectional run the inputs in two ways. Thus, the information preserved from the future and using the two hidden states combined that were able at any point in time to preserve information from both past and future.

To enhance the performance, there are several suggestions for future works based on our experiment results. Since the dataset in our experiment is very imbalance, we can try balancing the dataset by manual annotation, using text augmentation, or using a particular dataset balancing algorithm. In terms of architecture, we can tuning the parameter such as batch size, dropout proportion, learning rate, epoch, etc. Furthermore, we can add an embedding layer like word2vec, FastText, etc. since

TABLE VIII
$F1 - Score$ RESULTS FOR ALL MODELS

Dataset	Model	$F1 - Score(\%)$
Standard	LSTM	61.65
	Bi-LSTM	70.41
	GRU	63.38
	Bi-GRU	**71.04**
	CNN	62.92
Lowercase with Punctuation	LSTM	62.08
	Bi-LSTM	68.92
	GRU	62.48
	Bi-GRU	**70.61**
	CNN	62.47
Lowercase without Punctuation	LSTM	60.58
	Bi-LSTM	66.29
	GRU	61.17
	Bi-GRU	**68.12**
	CNN	63.61
Lowercase and Clean	LSTM	60.52
	Bi-LSTM	66.65
	GRU	61.71
	Bi-GRU	**67.45**
	CNN	62.43

in this experiment, we only build embedding from each word in our dataset. In addition to architecture, we also can use other potential features such as character embedding for enhancing the performance of this NER system.

References

[1] P. Bellot, E. Crestan, M. El-Bèze, L. Gillard, and C. de Loupy, "Coupling named entity recognition, vector-space model and knowledge bases for trec 11 question answering track," *NIST SPECIAL PUBLICATION SP*, no. 251, pp. 398–406, 2003.

[2] V. Gupta and G. S. Lehal, "Named entity recognition for punjabi language text summarization," *International journal of computer applications*, vol. 33, no. 3, pp. 28–32, 2011.

[3] J. Jiang, "Information extraction from text," in *Mining text data*. Springer, 2012, pp. 11–41.

[4] B. Babych and A. Hartley, "Improving machine translation quality with automatic named entity recognition," in *Proceedings of the 7th International EAMT workshop on MT and other language technology tools, Improving MT through other language technology tools, Resource and tools for building MT at EACL 2003*, 2003.

[5] E. F. Tjong Kim Sang, "Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition," in *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*, 2002. [Online]. Available: https://www.aclweb.org/anthology/W02-2024

[6] M. Collins and Y. Singer, "Unsupervised models for named entity classification," in *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

[7] E. F. Sang and F. De Meulder, "Introduction to the conll-2003 shared task: Language-independent named entity recognition," *arXiv preprint cs/0306050*, 2003.

[8] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," *arXiv preprint arXiv:1603.01360*, 2016.

[9] J. P. Chiu and E. Nichols, "Named entity recognition with bidirectional lstm-cnns," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 357–370, 2016.

[10] A. S. Wibawa and A. Purwarianti, "Indonesian named-entity recognition for 15 classes using ensemble supervised learning," *Procedia Computer Science*, vol. 81, pp. 221–228, 2016.

[11] N. Banik and M. H. H. Rahman, "Gru based named entity recognition system for bangla online newspapers," in *2018 International Conference on Innovation in Engineering and Technology (ICIET)*. IEEE, 2018, pp. 1–6.

[12] A. Žukov-Gregorič, Y. Bachrach, and S. Coope, "Named entity recognition with parallel recurrent neural networks," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 69–74.

[13] N. Reimers and I. Gurevych, "Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging," *arXiv preprint arXiv:1707.09861*, 2017.

[14] T. Gui, R. Ma, Q. Zhang, L. Zhao, Y.-G. Jiang, and X. Huang, "Cnn-based chinese ner with lexicon rethinking." in *IJCAI*, 2019, pp. 4982–4988.

[15] R. Weischedel, S. Pradhan, L. Ramshaw, M. Palmer, N. Xue, M. Marcus, A. Taylor, C. Greenberg, E. Hovy, R. Belvin *et al.*, "Ontonotes release 4.0," *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*, 2011.

[16] G.-A. Levow, "The third international chinese language processing bakeoff: Word segmentation and named entity recognition," in *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, 2006, pp. 108–117.

[17] H. He and X. Sun, "A unified model for cross-domain and semi-supervised named entity recognition in chinese social media," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[18] Y. Zhang and J. Yang, "Chinese ner using lattice lstm," *arXiv preprint arXiv:1805.02023*, 2018.

[19] M. Gridach and H. Haddad, "Arabic named entity recognition: A bidirectional gru-crf approach," in *International Conference on Computational Linguistics and Intelligent Text Processing*. Springer, 2017, pp. 264–275.

[20] Y. Benajiba and P. Rosso, "Arabic named entity recognition using conditional random fields," in *Proc. of Workshop on HLT & NLP within the Arabic World, LREC*, vol. 8. Citeseer, 2008, pp. 143–153.

[21] F. Dugas and E. Nichols, "Deepnnner: Applying blstm-cnns and extended lexicons to named entity recognition in tweets," in *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, 2016, pp. 178–187.

[22] Y. Munarko, M. Sutrisno, W. Mahardika, I. Nuryasin, and Y. Azhar, "Named entity recognition model for indonesian tweet using crf classifier," 2018.

[23] D. H. F. Riyanti Kusumawati, Dr. Yudi Wibisono, "Nergrit corpus," https://github.com/grit-id/nergrit-corpus, 2013.

[24] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[25] R. Weischedel, M. Palmer, M. Marcus, E. Hovy, S. Pradhan, L. Ramshaw, N. Xue, A. Taylor, J. Kaufman, M. Franchini *et al.*, "Ontonotes release 5.0 ldc2013t19," *Linguistic Data Consortium, Philadelphia, PA*, vol. 23, 2013.

[26] W. Gunawan, D. Suhartono, F. Purnomo, and A. Ongko, "Named-entity recognition for indonesian language using bidirectional lstm-cnns," *Procedia Computer Science*, vol. 135, pp. 425–432, 2018.

[27] C. Lyu, B. Chen, Y. Ren, and D. Ji, "Long short-term memory rnn for biomedical named entity recognition," *BMC bioinformatics*, vol. 18, no. 1, p. 462, 2017.

[28] L. Smith, L. K. Tanabe, R. J. nee Ando, C.-J. Kuo, I.-F. Chung, C.-N. Hsu, Y.-S. Lin, R. Klinger, C. M. Friedrich, K. Ganchev *et al.*, "Overview of biocreative ii gene mention recognition," *Genome biology*, vol. 9, no. S2, p. S2, 2008.

[29] J.-D. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier, "Introduction to the bio-entity recognition task at jnlpba," in *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*. Citeseer, 2004, pp. 70–75.

[30] L. Ratinov and D. Roth, "Design challenges and misconceptions in named entity recognition," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, 2009, pp. 147–155.

[31] Z. Zhang, “Improved adam optimizer for deep neural networks,” in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*. IEEE, 2018, pp. 1–2.

[32] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.

[33] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[34] M. Hossin and M. N. Sulaiman, “A review on evaluation metrics for data classification evaluations,” vol. 5, pp. 1–11, 03 2015.