NAMED ENTITY RECOGNITION USING BIDIRECTIONAL LSTM-CRF METHODS IN INDONESIAN TEXT

Hadi Permana[1], Ken Kinanti Purnamasari[2]

[1,2]Indonesian Computer University
Jl. Dipati Ukur No.112-116, Lebakgede, Coblong, Kota Bandung, West Java 40132
E-mail : hadi.prmn@gmail.com[1], ken.kinanti@email.unikom.ac.id[2]

ABSTRACT

Named Entity Recognition (NER) or the introduction of named entities is one of part or task of natural language processing (nlp). The purpose of NER is to identify or classify an entity such as the name of a person, organization, time, location and something else in a text that is very useful in the case of information extraction. In this study, the method used was the bidirectional LSTM-CRF method. Bidirectional LSTM combines the previous context and the next context by processing data from two directions which are then classified using CRF. There are several processes carried out, namely preprocessing; tokenizing, features (initCap, allCap, allLower, digits, contains digits and punctuation) and then carried out training and testing from preprocessing data. Based on the results of testing using training data as many as 25.709 word and testing 9406 word, bidirectional LSTM-CRF method obtain an accuracy of 87.77%.

Keywords : Named Entity Recognition, NER, Bidirectional LSTM-CRF, Natural language Processing.

1. INTRODUCTION

Named Entity Recognition (NER) or named entities is one of part or task of natural language processing (nlp). The purpose of the NER is to identify or classify an entity instance names of people, organizations, time, location and something other entities in a text [1]. NER can be used in other cases of natural language processing, such as information extraction and automed query generation.

Research on NER has been carried out in Indonesian. One of them is research that use the HMM method in the case of automatic question generation [2] with the accuracy of 42.54%. The low accuracy in the study was duet o ambiguous words that were not detected.

Meanwhile, in the machine learning method there is a method that is proven to have the highest state-of-the-art performance in the case of NER [3], namely bidirectional LSTM-CRF. Bidirectional LSTM combines the previous context and the next context by processing data from two direction [1] which is further classifed using CRF [3]. In a study conducted by Guillaume Lample, et al. [3] compared two neural architectures in overcoming NER, namely stack-LSTM (S-LSTM) and bidirectional LSTM-CRF. The results of the comparison in the English dataset, S-LSTM geta n accuracy of 90.33% and bidirectional LSTM-CRF geta n accuracy of 90.94%. In another study conducted by T Anh Le, et al. [1] bidirectional LSTM-CRF method gets 87.17 % accuracy compared to NeuroNER method which gets 85.37% accuracy for Gareev's dataset. From the studies that have been done it is proven that bidirectional LSTM-CRF method has a fairly high accuracy. Therefore, in this study the bidirectional LSTM-CRF method will be used to handle cases in Indonesian texts.

Based on the description, in this study the implementation of the bidirectional LSTM-CRF method in the case of Named Entity Recognition in Indonesian texts with data limitations to be used in this study will be political news articles from various sources.

2. RESEARCH CONTENT

Describes named entity recognition, news, corpus, research methods, process flow, input data, tokenization, feature, algorithm long short term memory (LSTM), bidirectional LSTM, conditional random field and test results.

2.1 Named Entity Recognition

Named entity recognition or NER is one of the tasks of natural language processing [1] which aims to recognize units of information such as names including people's names, organizations, and location names, and numerical expressions including

time, date, and expression of percent [1]. So, named entity recognition aims to identify any entity that has a name information.

2.2 News

News is a report o nan event or the latest event (actual), reports on facts that are actual, interesting, considered important, or extraordinary [8]. In this study, the news used is specific to a particular category, the political news. It because the NER system (Named entity recognition) in a previous study using the news with political categories.

2.3 Korpus

Corpus according to the Kamus Besar Bahasa Indonesia (KBBI) is a collection of written or oral utterrances used to support or test hypotheses about the structure of language. Corpus can also be interpreted as data used as a source of research material.

In this study the corpus used is the result of Rusliani [2] and Fachri's research [5].

2.4 Research Methods

The research methodology used is descriptive method [6]. There are four stages of research workflows, namely literature studies, dataset collection, software development, and testing. The following is the research workflow used.

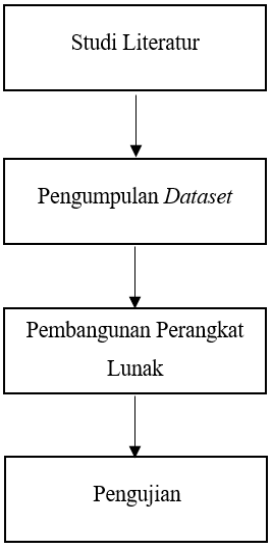

Figure 1. Research Flow

2.5 Process

There is a series of system processes that exist in the construction of the NER system. The following is the process flow in the NER system using bidirectional LSTM-CRF.

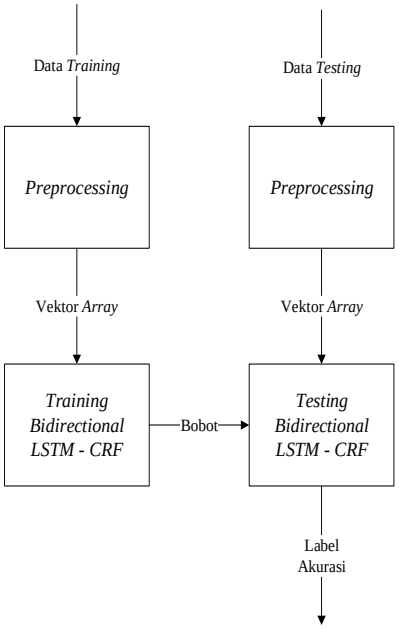

Figure 1. Process flow

Process flow in Figure 3. Begin by entering a dataset in the from of a news article that has been saved with the file format .txt, in the preprocessing stage there are two processes namely word tokenization and feature. The word tokenization process is used to change from a word sequence to a word token. The feature process is used to recognize the characteristics of a word token into a vector so that it can be processed by an algorithm. The training and testing process uses bidirectional long short term memory(Bi-LSTM) with the output layer using conditional random field (CRF). Input to Bi-LSTM form of vectors of a token that has been given features. The training process in Bi-LSTM-CRF produces the weights that will be used in the testing process. The testing process will get the label of each word token and the accuracy of the method used.

2.6 Input Data

The input data used in the NER process are training data and testing data taken from various online news sites such as kompas.com, detik.com and cnnindonesia.com. The data from the various sites combined is saved as a file .txt. The following is an example of the input data used in this study.

Terpianda mati kasus narkoba yang merupakan warga negara Perancis, Serge Atlaoui, mengajukan upaya hukum lanjutan berupa peninjauan kembali ke Pengadilan Negeri Tangerang. Sebelumnya, terpianda mati asal Filipina, Mary Jane Fiesta Veloso, juga telah mengajukan langkah serupa.

Figure 2. Sample Input Data

2.7 Preprocessing

In the preprocessing stage is done tokenization said then done giving the word to get its feature vector words. Following is the block diagram of the data process training.

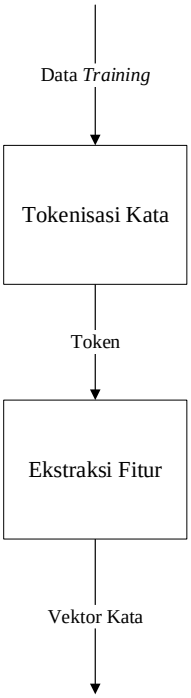

Figure 3. Preprocessing training

Similar to training data, the data is tokenized and then the feature is given to get the word vector. The following is a block diagram of preprocessing testing data.

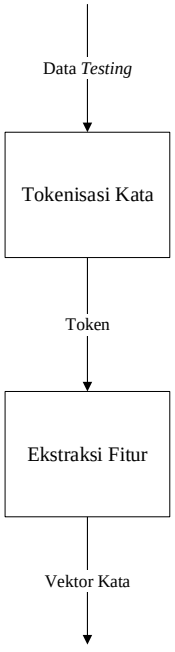

Figure 4. Preprocessing testing

2.8 Tokenization

Tokenization of words is the stage of separation of text into word per word using a separator. In the separation of words in this study using library nltk.word_tokenize. The steps carried out by library are as follows.

1. Separating token with space or tab
2. Character punctuation regarded as separate tokens
3. Word and punctuation are considered one token if it is followed by another word, eg: S.T, 2.3, etc.

Here is the word has been tokenize with the library.

Table 1. Tokenization

No	Token	No	Token
1	Terpianda	23	Negeri
2	Mati	24	Tangerang
3	kasus	25	.
4	narkoba	26	Sebelumnya
5	yang	27	,
6	merupakan	28	Terpianda
7	warga	29	Mati
8	negara	30	asal
9	Perancis	31	Filipina
10	,	32	,
11	Serge	33	Mary
12	Atlaoui	34	Jane
13	,	35	Fiesta
14	mengajukan	36	Veloso
15	upaya	37	,
16	hukum	38	juga
17	lanjutan	39	telah
18	berupa	40	mengajukan
19	peninjauan	41	langkah
20	kembali	42	serupa
21	Ke	43	.
22	Pengadilan		

2.9 Feature

Every word or token is converted into a vector that can be read by the method used. The feature used is spelling feature [4] by taking six features based on the consideration of the dataset used. The following are the features that are used.

Table 2. Fitur

No	Fitur	Keterangan
1	InitCap	Identify each token whose letter begins with capital.
2	AllCap	Identify each token for which all letters are capital.
3	AllLower	Identify each token that is all lowercase.
4	Digits	Identify all tokens that are all digits.
5	ContaintsDigits	Identify each token containing digits.

6	Punctuation	Identify each token containing punctuation.

Each feature in table 2 will give a value of 1 or 0.

1. InitCap will be worth 1 if the token has a capital letter at the beginning of the word and if not it will be 0. Example “Seorang”,”Pengamat”, etc.
2. AllCap will be worth 1 if the token contains all capital letters andi f not it will be 0. For example “NI”, “IBM”, etc.
3. AllLower will be worth 1 if the token contains all the small letters all and if not it will be 0. For example “tahun”, “yang”, etc.
4. Digits will be worth 1 if the token is a number and if it not will be 0. For example “2014”, “21”, etc.
5. Combined letters and numbers worth 1 if the token has letters and numbers if it not will be 0.
6. Punctuation is worth 1 if a token has a punctuation mark and if it is not will be 0. For example “24-Aug”.

Table 3. Example Feature

No	Word	Feature					
		F1	F2	F3	F4	F5	F6
1	Terpianda	1	0	0	0	0	0
2	mati	0	0	1	0	0	0
3	kasus	0	0	1	0	0	0
4	narkoba	0	0	1	0	0	0
5	yang	0	0	1	0	0	0
6	merupakan	0	0	1	0	0	0
7	warga	0	0	1	0	0	0
8	negara	0	0	1	0	0	0
9	Perancis	1	0	0	0	0	0
10	,	0	0	0	0	0	1
11	Serge	1	0	0	0	0	0
12	Atlaoui	1	0	0	0	0	0
13	,	0	0	0	0	0	1
14	mengajukan	0	0	1	0	0	0
15	upaya	0	0	1	0	0	0
16	hukum	0	0	1	0	0	0
17	lanjutan	0	0	1	0	0	0
18	berupa	0	0	1	0	0	0
19	peninjauan	0	0	1	0	0	0
20	kembali	0	0	1	0	0	0
21	ke	0	0	1	0	0	0
22	Pengadilan	1	0	0	0	0	0
23	Negeri	1	0	0	0	0	0
24	Tangerang	1	0	0	0	0	0
25	.	0	0	0	0	0	1

2.10 Long Short Term Memory (LSTM)

Recurrent neural networks (RNN) is a method that operates for data sequences. This method takes input from the sequence vector (x_1, x_2, ..., x_n) adn becomes another sequence (h_1, h_2, ..., h_n) [3]. The RNN method cannot be used in long-term dependencies which creates a vanishing gradient problem [1]. Therefore long short term memory (LSTM) was developed to overcome the problem of vanishing gradient [1]. LSTM replaces hidden units on the RNN architecture with a unit that can be called a memory block consisting of four component: input gate, output gate, forget gate and memory cell [3]. The formulas of the four components are as follows.

$$i_t = \sigma (W_i x_t + U_i h_{t-1} + b_i) \quad (1)$$

Input gate (i_t) used to change values in the cell state(c_t) whit W_i and U_i are matrix weights multiplied by vectors x_t and $h_{t-1.}$

$$f_t = \sigma (W_f x_t + U_f h_{t-1} + b_f) \quad (2)$$

Forget gate (f_t) is used to delete information from the cell state(c_t). Information from the previous hidden state (h_{t-1}) and the input (x_t) is calculated by the sigmoid function (σ) with an output value between 0 and 1. If the output value approaches 0, the information will be deleted, and if the output value approaches 1, the information will be stored.

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh (W_c x_t + U_c h_{t-1} + b_c) \quad (3)$$

Previous cell state (c_{t_1}) multiplied ($\odot$) with forget gate (f_t), there is a possibility that the value of the product will decrease if multiplied by the value (f_t) which is close to 0. Then the value of i_t multiplied ($\odot$) with the current cell state function to get the current cell state value with a value between 1 and -1.

$$o_t = \sigma (W_o x_t + U_o h_{t-1} + b_o) \quad (4)$$

Same is the case with input gate(i_t), output gate (o_t) is used to determine the value of the new hidden state (h_t).

$$h_t = o_t \odot \tanh(c_t) \quad (5)$$

Value of output gate (o_t) multiplied by the result of the tanh function for cell state(c_t). Which new hidden state and new cell state will be used for calculation in the next step(t).

Where σ is a sigmoid function with an equation

$$f(x) = \frac{1}{1+\exp(-x)} \quad (6)$$

And tanh with equations

$$\tanh(x) = \frac{2}{1+\exp(-2x)} - 1 \quad (7)$$

2.11 Bidirectional LSTM

Bidirectional LSTM utilizes the previous context and next context by processing data from two directions with separate hidden layers [10] [1]. Forward layer to represent the previous context, and backward layer to represent the context afterwards [1]. Output from a combination of two-way hidden layer $\vec{h}_t$ and $\overleftarrow{h}_t$ are: $y_t = W_{\vec{h}y}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t$.

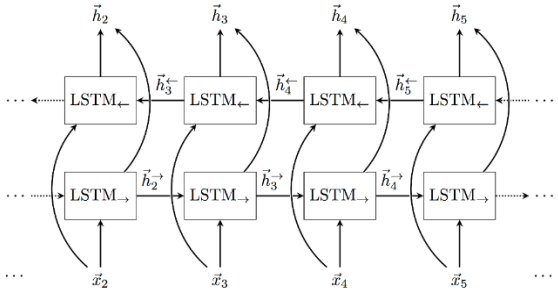

Figure 5. Bidirectional LSTM models

2.12 Conditional Random Field(CRF)

Conditional Random Field is a probabilistic model used to predict structured data that has been used in various tasks, such as computer vision, natural language processing.

The CRF model conducts training to predict a vector y {y0,y1,y2,..,yT} from a sentence X {x0, x1, x2,..xT} with

$$p(y|x) = \frac{e^{score(x,y)}}{\sum_{y'} e^{score(x,y')}} \quad (8)$$

Where to look for $score(x, y)$ can use

$$score(x, y) = \sum_{i=0}^{T} A_{yi,yi+1} + \sum_{i=1}^{T} P_{i,yi} \quad (9)$$

Where $A_{yi,yi+1}$ is the emission probability that represents the transition score from the i tag to the j tag. $P_{i,yi}$ is the transition probability that represents the transition score of the tag j to the word i.

2.13 Bidirectional LSTM-CRF

Bidirectional LSTM-CRF is a combination of bidirectional LSTM methods and CRF models. Word vectors are calculated bidirectional LSTM to produce a score that represents the possibility of tags in each word in the sentence. That means $P_{i,yi}$ from equation 9 is replaced by the result of Bidirectional LSTM [1]. So that the CRF model only counts $A_{yi,yi+1}$.

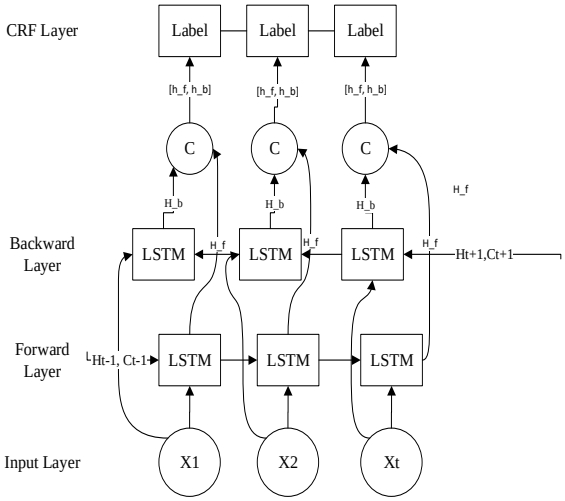

Figure 6. Bidirectional LSTM-CRF Models

2.14 Test Result

The accuracy test results obtained from two test scenarios, namely changing the number of epochs and changing the learning rate value. Here are the results of testing carried out on data accuracy testing.

2.14.1 Testing Scenario 1

In testing scenario 1, method testing is done by converting the epoch value to 40, and the learning rate value is 0.01, 0.015, 0.02 and 0.001.

a. Epoch 40 and learning rate 0.01

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 4. Prediction Results

Name Entiy	System Results	Right Prediction
Person	239	669
Organization	73	296
Time	237	254
Quantity	38	42
Location	112	183
Other	7448	7962
Total	**8147**	**9406**

From the following table the accuracy obtained can be calculated by dividing the total results of the system with the correct total predictions, then the results for those multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 40 and the learning rate to 0.01 is 86.61%

b. Epoch 40 and learning rate 0.015

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 5. Prediction Results

Name Entiy	System Results	Right Prediction
Person	274	669
Organization	53	296
Time	236	254
Quantity	38	42
Location	105	183
Other	7502	7962
Total	**8208**	**9406**

From the following table the accuracy obtained can be calculated by dividing the total results of the system with the correct total predictions, then the results for those multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 40 and the learning rate to 0.015 is 87.26 %.

c. Epoch 40 and learning rate 0.02

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 6. Prediction Results

Name Entiy	System Results	Right Prediction
Person	258	669
Organization	60	296
Time	235	254
Quantity	38	42
Location	103	183
Other	7507	7962
Total	**8201**	**9406**

From the following table the accuracy obtained can be calculated by dividing the total results of the system with the correct total predictions, then the results for those multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 40 and the learning rate to 0.02 is 87.18 %

d. Epoch 40 and learning rate 0.001

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 7. Prediction Results

Name Entiy	System Results	Right Prediction
Person	205	669
Organization	42	296
Time	236	254
Quantity	38	42
Location	95	183
Other	7587	7962
Total	**8203**	**9406**

From the following table the accuracy obtained can be calculated with equations dividing the total results of the system with the correct total predictions, then the quotient is multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 40 and the learning rate to 0.01 is 87.21%. The following is a table from the test results for scenario 1.

Table 8. Testing Scenario 1

Epoch	Learning rate	Accuracy(%)
40	0.01	86.61
40	0.015	87.26
40	0.02	87.18
40	0.001	87.21

Based on the results of the scenario 1 test, it can be concluded that the epoch value and the learning rate that have the highest accuracy are the epoch 40 value and the learning rate value 0.015 with an accuracy of 87.26 %.

2.14.2 Testing Scenario 2

In testing scenario 2, method testing is done by converting the epoch value to 50, and the learning rate value to 0.01, 0.015, 0.02 and 0.001.

a. Epoch 50 and learning rate 0.01

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 9. Prediction Results

Name Entiy	System Results	Right Prediction
Person	338	669
Organization	57	296
Time	232	254
Quantity	38	42
Location	99	183
Other	7478	7962
Total	**8242**	**9406**

From the following table the accuracy obtained can be calculated by dividing the total results of the system with the correct total predictions, then the results for those multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 50 and the learning rate to 0.01 is 87.62%.

b. Epoch 50 and learning rate 0.015

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 10. Prediction Results

Name Entiy	System Results	Right Prediction
Person	328	669
Organization	58	296
Time	236	254
Quantity	38	42
Location	96	183
Other	7457	7962
Total	**8213**	**9406**

From the following table the accuracy obtained can be calculated by dividing the total results of the system with the correct total predictions, then the results for those multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 50 and the learning rate to 0.015 is 87.32%.

c. Epoch 50 and learning rate 0.02

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 11. Prediction Results

Name Entiy	System Results	Right Prediction
Person	327	669
Organization	61	296
Time	234	254
Quantity	38	42
Location	102	183
Other	7491	7962
Total	**8253**	**9406**

From the following table the accuracy obtained can be calculated by dividing the total results of the system with the correct total predictions, then the results for those multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 50 and the learning rate to 0.02 is 87.74 %.

d. Epoch 50 and learning rate 0.001

The following are the results of system predictions from six entities, namely person, organization, time, quantity location and other.

Table 12. Prediction Results

Name Entiy	System Results	Right Prediction
Person	217	669
Organization	40	296
Time	237	254
Quantity	38	42
Location	93	183
Other	7631	7962
Total	**8256**	**9406**

From the following table the accuracy obtained can be calculated by dividing the total results of the system with the correct total predictions, then the results for those multiplied by one hundred percent. Then the accuracy of the test by changing the epoch value to 50 and the learning rate to 0.001 is 87.77%. The following is a table from the test results for scenario 2.

Table 13. Testing Scenario 2

Epoch	Learning rate	Accuracy(%)
50	0.01	87.62
50	0.015	87.31
50	0.02	87.61
50	0.001	87.77

Based on the results of testing scenario 2, it can be concluded that the epoch and learning rate values that have the highest accuracy are epoch 50 and learning rate 0.001 with an accuracy of 87.77%.

3. CONCLUSION

Based on the results of two test scenarios that have been carried out, it can be concluded that the accuracy of the method bidirectional LSTM-CRF on Indonesian NER system is 87,77%. Accuracy is obtained from the arrangement of parameters that have epoch limits 50, learning rate of 0.001.

The suggestion for further research is to add data with entity name, organization, time and quantity because in the dataset used almost 80% is data with other entities. Extra features can also be done in order to detect entities with more accuracy.

BIBLIOGRAPHY

[1]G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami and C. Dyer, "Neural Architectures for Named Entity Recognition," arxiv:1603.01360v3, 2016.

[2]A. L. T., A. M. Y. and B. M. S., "Application of a Hybrid Bi-LSTM-CRF model to the task of Russian Named Entity Recognition," 2017.

[3]Rusliani, "Named Entity Recognition Pada Teks Berbahasa Indonesia Untuk Pembangkit Pertanyaan Otomatis," UNIKOM, Bandung, 2017.

[4]Z. Huang, W. Xu and K. Yu, "Bidirectional LSTM-CRF Models for Sequence Tagging," arxiv:1508.01991v1, 2015.

[5]M. Fachri, "Pengenalan Entitas Bernama Pada Teks Bahasa Indonesia Menggunakan Hidden Markov Model," Universitas Gadjah Mada, Yogyakarta, 2014.

[6]Z. A. Hasibuan, Metodologi Penelitian Pada Biandg Ilmu Komputer And Teknoogi Informasi, Depok: Universitas Indonesia, 2007.

[7]R. S. Pressmann, Software Engineering, Yogyakarta: Andi, 2010.
[8]K. Budiman, “Dasar - Dasar Jurnalistik,” dalam Pelatihan Jurnalistik, Jawa, Info Jawa, 2005, pp. 1 - 4.
[9]F. Amin, “Sistem Temu Kembali Informasi dengan Metode Vector Space Model,” Jurnal Sistem Informasi Bisnis, vol. 02, pp. 78-83, 2012.
[10]M. Maimaiti, A. Wumaier, K. Abiderexiti and T. Yibulayin, “Bidirectional Long Short-Term Memory Network with a Conditional Random Field Layer for Uyghur Part-Of-Speech Tagging,” 2017.
[11]D. Bell, “An introduction to the Unified Modeling Language,” IBM, 15 June 2003. [Online]. Available: https://www.ibm.com/developerworks/rational/library/769.html. [Diakses 4 October 2018].
[12]S. W. Ambler, “Introduction to the Diagrams of UML 2.X,” Agile Modeling, [Online]. Available: http://www.agilemodeling.com/essays/umlDiagrams.htm. [Diakses 4 October 2018].
[13]F. A. Aslam and H. N. Mohammed, “Efficient Way Of Web Development Using Python And Flask,” International Journal of Advanced Research in Computer Science, vol. 6, no. 2, pp. 54-57, 2015.
[14]I. A. Diana, Sistem Komunikasi, Bandung: Universitas Pendidikan Indonesia, 2012.
[15]K. Xu, Z. Zhou, T. Hao and W. Liu, “A Bidirectional LSTM and Conditional Random Fields Approach to Medical Named Entity Recognition,” 2017.
[16]Z. Liu, B. Tang, X. Wang and Q. Chen, “De-identification of clinical notes via reccurent neural network and conditional random field,” Journal of Biomedical Informatics, 2017.
[17]A. Solihin, PEMROGRAMAN WEB DENGAN PHP AND MYSQL, Penerbit Budi Luhur, 2016.
[18]K. Xu, Z. Zhou, T. Hao and W. Liu, “A Bidirectional LSTM and Conditional Random Fields Approach to Medical Named Entity Recognition,” 2018.
[19]A. Setioaji, L. Muflikhah and M. A. Fauzi, “Named Entity Recognition Menggunakan Hidden Markov Model and Algoritma Viterbi pada Teks Tanaman Obat,” Jurnal Pengembangan Teknologi Informasi and Ilmu Komputer, vol. I, no. 12, pp. 1858-1864, 2017.
[20]N. Jaariyah, “Pengenalan Entitas Bernama pada Teks Bahasa Indonesia Menggunakan Conditional Random Fields,” Perpustakaan UNIKOM, Bandung, 2017.
[21]M. I. Tiarasani, “Pengenalan Entitas Bernama Pada Artikel Berita Berbahasa Indonesia Menggunakan Metode Hidden Markov Model And Rule Based,” UGM, Yogyakarta, 2018.
[22]I. Irfana, “Pengenalan Entitas Bernama Pada Artikel Berita Bahasa Indonesia Menggunakan Metode Berbasis Aturan,” UGM, Yogyakarta, 2015.