JOIN (Jurnal Online Informatika)
p-ISSN: 2528-1682, e-ISSN: 2527-9165
Volume 8 Number 2 | December 2023: 131-138
DOI: 10.15575/join.v8i2.898

Improving Indonesian Named Entity Recognition for Domain Zakat Using Conditional Random Fields

Nur Febriana Widiyanti[1], Husni Teja Sukmana[2*], Khodijah Hulliyah[3]
Dewi Khairani[4], Lee Kyung Oh[5]
[1,2,3,4]Department of Informatics, UIN Syarif Hidayatullah Jakarta, Indonesia
[5]Department of Computer Engineering, Sun Moon University, South Korea

Article Info

Article history:

Received August 10, 2022
Revised August 03, 2023
Accepted August 08, 2023
Published December 28, 2023

Keywords:
Named Entity Recognition
Conditional Random Fields
Natural Language Processing
Charity
Zakat

ABSTRACT

In Indonesia, where the majority of the population is Muslim, one of the obligations of a Muslim is zakat. To reduce illiteracy about zakat among Muslims, they need to have access to basic information about it. In order to facilitate the acquisition of this information, this study utilized named entity recognition (NER) and defined 12 named entity classes for the zakat domain, including the pillars of Islam, various types of zakat, and zakat management institutions. The Conditional Random Fields method was used for testing Indonesian-NER in three scenarios. In the specific context of the Zakat domain, NER can extract information about organizations, individuals, and locations involved in collecting and distributing Zakat funds. This information can improve the Zakat system's efficiency and transparency and support research and analysis on Zakat-related topics. The average performance evaluation of the Indonesian-NER model showed a precision of 0.902, recall of 0.834, and an F1-score of 0.867.

Corresponding Author:

Husni Teja Sukmana,
Informatics Department, Faculty of Science & Technology, UIN Syarif Hidayatullah Jakarta,
UIN Syarif Hidayatullah Jakarta,
Jl. Ir H. Juanda No.95, Cemp. Putih, Kec. Ciputat Tim., Kota Tangerang Selatan, Banten 15412
Email: husniteja@uinjkt.ac.id

1. INTRODUCTION

Named Entity Recognition (NER) is the first step towards information extraction that seeks to identify an entity mentioned in a text into a predetermined category. For example, the name of a person, organization, location, and something else in a text [1]–[3]. The main problem with Named Entity Recognition (NER) is ambiguous words or phrases, so to reduce this ambiguity, a BIO (Beginning, Inside, Outside) format is needed when labeling data [4][5]. The unclear boundary determination is another problem with Named Entity Recognition (NER). or punctuation of the phrase [6].

Named Entity Recognition (NER) is an approach that has rules which require manual definition to recognize the phrase. For example, the word or phrase after "at" is most likely a location, except in some instances. Manually defining it, however, requires more time and presents more significant difficulties in covering all cases, But it can clarify the determination of the limit or punctuation of the phrase to reduce ambiguity [6].

The methods developed for Named Entity Recognition with machine learning include Hidden Markov Model (HMM) [7], Decision Tree [8], Maximum Entropy [9], Support Vector Machine (SVM) [10], Conditional Random Fields (CRF) [11], and Association rules mining [12], [13]. Current research on Indonesian Language for accuracy in Named Entity Recognition (NER) is still not maximized [14] due to the nature of the Indonesian Language because Indonesian has unique orthographic, morphological and

p-ISSN: 2528-1682
e-ISSN: 2527-9165

local contextual characteristics, both formal and informal [15]. So managing Named Entity Recognition (NER) for Indonesians is challenging and complex.

One potential problem this study could address is the difficulty of extracting structured information from unstructured text. NER systems often have to deal with variations in language use, including different spellings, abbreviations, and contexts that can make it challenging to accurately identify and classify named entities in the Zakat domain. Not much research has been identified in Indonesia regarding the NER in Indonesian [10] compared to English [15]. Some issues related to Indonesian NER can be found in [12]. Zakat can be categorized as a specific domain [4], thus, the chance for making or increasing the performance of this area is still vast. Research related to Named Entity Recognition (NER) in Indonesian for the zakat domain has been carried out by [4] by using *stanford-NER* approach. These NER testing results are about 0.7641 for precision, 0.4544 for recall, and the F1-score is about 0.5655.

From the previous studies related to zakat, the model's evaluation results were still not maximized, so we want to modify the model to improve it. In this study, we developed a model of Named Entity Recognition in Indonesian for the zakat domain using a different method, namely the Conditional Random Fields (CRF) method. Therefore, the researcher wants to know how well the model uses Conditional Random Fields (CRF) to recognise entities for the Indonesian language zakat domain. With the Indonesian-NER for the zakat domain, it is hoped that it can provide fundamental knowledge or information about zakat in Indonesia to increase Muslim literacy regarding zakat [16]. This study aims to develop a model that can accurately identify and extract named entities (such as people, organizations, and locations) from text within the context of the Zakat domain. The study involves collecting and annotating a text dataset, training and testing the NER system on this dataset, and evaluating the system's performance using precision, recall, and F1-score metrics.

2. METHOD

We follow at least eight steps for making NER in Zakat, as shown in Figure 1 for the research flow. We start with data processing by using web scraping. Web scraping is a technique used to get information from websites, and web scraping will extract data automatically without copying it manually [17]. After the data is obtained, the next stage is tokenization. Tokenization is the stage of separating text that is obtained from a sentence, paragraph or document, then used as specific tokens. This process is done using the anaconda prompt**.**

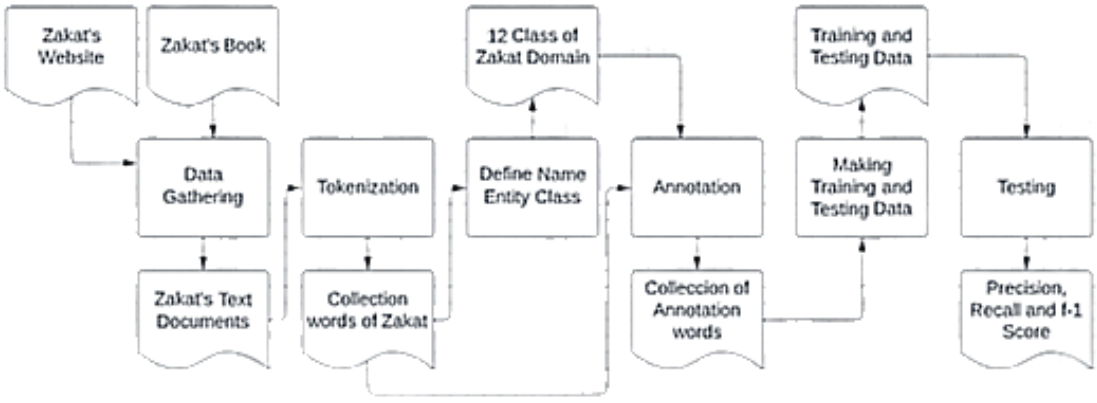

Figure 1. Research Flow

Tokenization can be challenging due to the complexity of natural language and the various ways words and punctuation marks can be combined and written. For example, words can be written with or without spaces between them, and punctuation marks can be used in different ways to convey meaning or structure; to address these challenges, tokenization algorithms may use a combination of rules and machine learning techniques to identify and split the text into tokens accurately.

After tokenization, the next step is to map the class entities and entities for the zakat domain based on the analysis by researchers obtained from the results of data preparation. Defining entity classes is an important aspect of designing and building an NER system, as it determines the types of entities that the system will be able to recognize and extract from text. The process of identifying and classifying the types of entities that a Named Entity Recognition (NER) system should be able to identify

and extract from text. Entity classes are categories of entities that share similar characteristics or belong to the same domain. Some common entity classes include people, organizations, locations, and events.

Defining classes in a classification system is crucial for several reasons, primarily because it delineates the extent and capabilities of the classification system. By explicitly outlining the types of entities or items that the system should recognize and classify, one can guarantee the accurate and effective execution of its intended tasks. Defining classes facilitates the creation of a dataset comprising annotated text or examples, where entities or items are manually labeled with their respective classes. This dataset serves the dual purpose of training the classification system and assessing its performance, ensuring consistency and accuracy in the system's assignments. Furthermore, specifying the unique characteristics and features associated with each class aids the system in acquiring the ability to identify and classify entities or items correctly.

To define entity classes, it is often necessary to create a list of the types of entities that the NER system should be able to recognize. This list can be based on domain-specific knowledge or the types of entities commonly found in the text the system will process. Once the entity classes have been defined, the NER system can be trained on a dataset of annotated text, where the entities have been manually labeled with their corresponding entity classes.

In this study, we have trained a model with the implementation of conditional random fields provided by sklearn-crfsuite. The data distribution in the form of samples into data training and data testing in this study uses the scikit-learn library. The division of the dataset portion into two parts, training and testing using the train-test split method in the python library. This scenario aims to test a model in different dataset states. In this study,we used three scenarios, namely 20% data testing and 80% data training, 30% data testing and 70% data training, and 40% data testing and 60% data training.

We are testing the performance of the Indonesian-NER model in this study by measuring the value of precision, recall and F1-score. These metrics are usually used to evaluate the performance of a classification system, such as a Named Entity Recognition (NER) system. These metrics quantify the system's accuracy and effectiveness in identifying and classifying entities in text.

Precision measures the proportion of true positive predictions made by the classification system, relative to the total number of positive predictions. A high precision indicates that the system is good at identifying true positive instances and not making false positive predictions. Recall measures the proportion of true positive instances that the classification system identified relative to the total number of positive instances in the dataset. A high recall indicates that the system is able to identify most or all of the positive instances.

The F1-score combines precision and recall and is calculated as the harmonic mean of the two metrics. The F1-score is a helpful metric for comparing the performance of different classification systems, as it considers both the system's precision and recall. In the context of NER, precision, recall, and F1-score can be used to evaluate the system's performance in identifying and classifying named entities in text. By measuring the NER system's precision, recall, and F1-score, you can gain insights into the strengths and weaknesses of the system and make improvements as needed. Here is the formula for the metrics:

$$\text{Precision} = (\text{True Positives}) / (\text{True Positives} + \text{False Positives}) \quad (1)$$

$$\text{Recall} = (\text{True Positives}) / (\text{True Positives} + \text{False Negatives}) \quad (2)$$

$$\text{F1-score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (3)$$

In these formulas, "True Positives" refers to the number of instances correctly identified by the classification system as belonging to the positive class. "False Positives" refers to the number of instances incorrectly identified by the system as belonging to the positive class. "False Negatives" refers to the number of instances incorrectly identified by the system as belonging to the negative class.

3. RESULTS AND DISCUSSION

This research data sourced from the internet was obtained through web scraping techniques. Websites for web scraping are the BAZNAS website, Rumah Zakat website, and Dompet Dhuafa's website. The data is extracted from data form (pdf) into text format. In this study, the author only uses

text data from the e-book "Zakat and Waqf Law" sourced from the Ministry of Religious Affairs of the Republic of Indonesia. To transform the data form (pdf) into txt form, the author uses https://pdftotext.com/. After the data collected, we perform the flow of our research from tokenization to model evaluation.

3.1. Tokenization

The result of this tokenization step is a collection of zakat's words (23,196 words), as we have depicted in Table 1. There are various approaches to tokenization, and the choice of approach may depend on the specific requirements and constraints of the NER system. Some standard tokenization methods include word-level tokenization, which divides the text into individual words, and character-level tokenization, which divides the text into individual characters [18].

Table 1. Tokenization Example

Document	Tokenization			
In terms, zakat comes from Arabic, zakah, or zakat, which means certain assets that must be issued by Muslim people and given to groups who have the right to receive it, the poor, the poor and so on.	In terms , zakat comes from Arabic , zakah or zakat	, which means certain assets that must be issued by people	who are Muslim and given to groups who have the right	to receive it , the poor , and so on .

3.2. Defining Classes

After the tokenization, we define classes; in this study 12 classes were defined by entities for the zakat domain, namely the pillars of Islam, zakat, types of zakat, zakat mal, mustahik, muzzaki, zakat management institutions, amil zakat institutions, sharia zakat, zakat nisab, zakat rates and person. Table 2 describes the example of the named entity classes and its example.

Table 2. The named entity class list

No	Named entity *class*	Label	Named entity example
1	Pillars of islam	RIM	Syahadat, salat/shalat, fasting, zakat, hajj
2	Zakat	ZAK	Types of zakat, mustahik /mustahiq /zakat recipients, muzzaki/muzaki, zakat management institutions/zakat management, shari'a zakat
3	Types of zakat	JZT	Zakat fitrah, zakat mal/zakat maal
4	Zakat mal	MAL	Income zakat, trade zakat, gold zakat, silver zakat, savings zakat, asset rental zakat, professional zakat, deposit zakat, company zakat, agricultural zakat, livestock zakat, stock zakat, gift zakat
5	Mustahik	MUS	Fakir, poor, amil/zakat administrator, gharimin/ghorimin, fisabilillah, ibn sabil, riqab, mualaf, ashnaf
6	Muzzaki	MUZ	Pay zakat, rich people, well-off people, people who pay tithe
7	Zakat management institutions	LPZ	BAZNAS/national amil zakat agency, baz/zakat amil agency, LAZ/amil zakat institution
8	LAZ/ amil zakat institutions	LAZ	Zakat house, dhuafa wallet, amil zakat institution daarut tauhid
9	Shari'a zakat	SYZ	Nisab of zakat, rates of zakat /level of zakat
10	Nisab of zakat	NSB	85 grams of pure gold, 595 grams of pure silver, 653 kg of grain, 20 dinars, 200 dirhams, 40 dirhams, 5 wasaq, 40 heads, 5 heads, 30 heads, 20 misqals, 10 heads, 15 heads. 20 head, 25 head, 36 head, 46 head, 50 head, 100 head, 120 head, 200 head, 300 head
11	Rates of zakat	TAZ	2.5%, 2.5 kg, 3.5 liters, 20%, one sha, 1 head, 2 heads, 3 heads, 4 heads, 10%, 5%
12	Persons	PER	Muslim, soul, a muslim, believers, yusuf qardhawi, jumhur ulama, abu ja'far al-baqir, hasan, mujahid, fuquha syafi'iyah, abu hanifah, imam hanafi, imam maliki, imam shafi'i, imam hambali, sadono soekirno, hadi permono, sayyid sabiq, ibn faris, mundzir qahf, imam ghazali, imam Syatibi, sheikh ulaith, sheikh mahmud syaltut, sheikh abu zahrah, ibn arabi, abu muhammad ibn qutaibah

This allows the system to learn to identify and classify entities based on the characteristics and patterns specific to each entity class.

3.3. Annotation

As mentioned before, in this study, we use a data set consisting of 23,196 words that are annotated manually. In document annotations, word labelling is carried out using the BIO format, which stands for Beginning (B), Inside (I) and Outside (O). In the general annotation format to mark an entity, in the BIO format where "B" is the beginning of the entity, "I" is an intermediate entity or is still related to the entity, and "O" is a word that does not include a named entity [19]. The result of the manual annotation example is shown in Table 3.

Table 3. Example of a named entity manual annotation

Entity	Named-entity manual annotation
national	B-LPZ
amil	I-LPZ
zakat	I-LPZ
agency	I-LPZ
gives	O
zakat	B-JZT
fitrah	I-JZT
to	O
fakir	B-MUS
poor	B-MUS

In Table 3, we have identified four entities in the sentence: national amil zakat agency, zakat fitrah, fakir and poor. *"The national zakat agency"* is a phrase with 4 syllables marked with B-LPZ and I-LPZ (See Table 2). Zakat fitrah is a phrase with 2 syllables marked with B-JZT and I-JZT, with this labeling we can see whether it is a word or a phrase. With the BIO format, it can reduce ambiguous words or phrases, for example the sentences of the national amil zakat agency and zakat fitrah, both sentences have the same word "zakat" but have different classes (also different meanings). Conditional Random Fields (CRF) consider the labels that appear before and each feature is assigned a weight to calculate the probability of the next label [6].

When defining an entity class for the zakat domain, the entities are identified and semantically classified into the marked class. The class entities taken in this study are related to the zakat domain (12 classes). They consist of Islam, zakat, types of zakat, zakat mal, mustahik, muzzaki, zakat management institutions, amil zakat institutions, shari'a zakat, nisab of zakat, rates of zakat, and persons. Table 3 explains the classes, and their entity example from a collection of zakat's words. We put labels for each named entity class that we have defined based on their abbreviation.

3.4. Training

We trained a model with the implementation of the Conditional Random Field provided by sklearn-crfsuite. Initialize the model instance and fit the training data with the fit method. The following is a script to train the Indonesian-NER model.

```
crf = sklearn_crfsuite.CRF(algorithm='lbfgs',
    c1=0.1,
    c2=0.1,
    max_iterations=100,
    all_possible_transitions = True
    )
crf.fit(X_train, y_train)
```

Please note that this method randomly divides the train set and test set. So, if we repeat the running process, then of course, the results obtained will change. Therefore, the results of the running evaluation do not change on *all_possible_transitions* = True.

3.5. Testing

In this study, the performance testing of the Indonesian-NER model was carried out with ten experiments or scenarios using a train/test split. The model evaluation in this study focuses on the three best scenarios, namely test data of 20%, 30%, and 40% of the entire dataset displayed in Table 4. Based on our evaluation, we can conclude that the model worked well, as displayed in Figure 2. All results are above 80% compared to a previous study [4]. In the previous study, the model evaluation concludes the precision, recall, and F1-score values of 0.7641, 0.4544, and 0.5655, respectively.

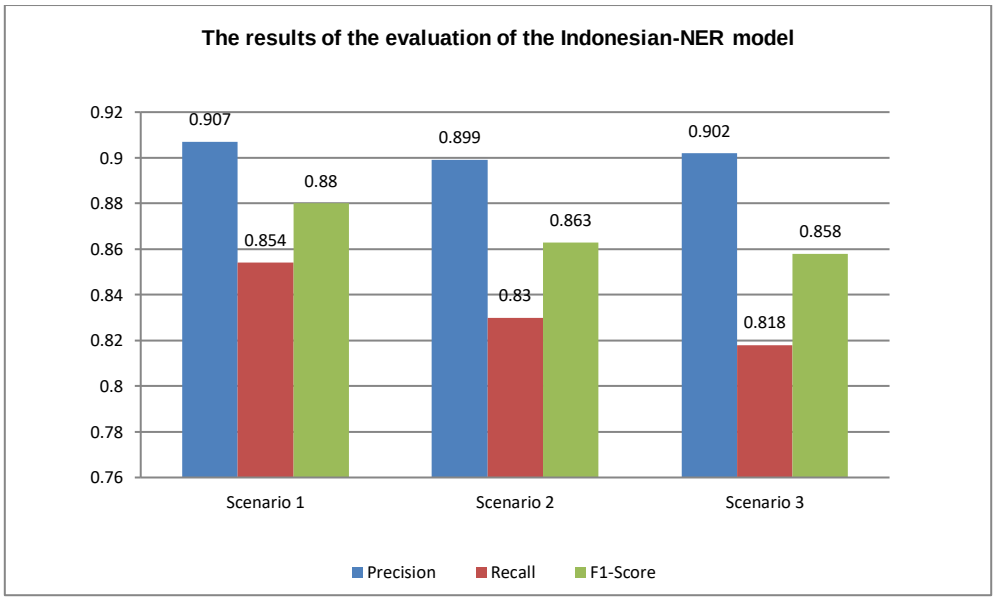

Figure 2. The results of the evaluation of the Indonesian-NER model

Our explanation for this increase is that our model used more data in this experiment which is consistent with the previous research [20], [21]. For several reasons, using more data in training can result in a more accurate model. First, a more extensive dataset allows the model to learn from more diverse and representative examples, which can help it generalize better to new data. For example, if the model is trained on a large and diverse dataset, it may recognize better named entities in different contexts and variations rather than just those seen in the training data. Second, a larger dataset can allow the model to learn more subtle and complex patterns and relationships in the data, which can improve its accuracy. For example, a model trained on a large dataset may be able to learn more fine-grained distinctions between different named entities or recognize more subtle cues and features that are indicative of a particular named entity.

Finally, a larger dataset can give the model more opportunities to practice and improve its performance. The more examples the model sees during training, the more chances it has to learn and improve its accuracy. Overall, using more data in training can help improve a model's accuracy by providing a more diverse and representative set of examples, allowing it to learn more complex and subtle patterns in the data and giving it more opportunities to practice and improve its performance.

Table 4. The average results of the evaluation of the Indonesia-NER model

Scenario	Precision	Recall	F1-Score
80% training & 20% testing	0.907	0.854	0.880
70% training & 30% testing	0.899	0.830	0.863
60% training & 40% testing	0.902	0.818	0.858
Average	**0.902**	**0.834**	**0.867**

4. CONCLUSION

The problem addressed in this research is the accuracy and efficiency of Named-Entity Recognition (NER) in Indonesia's zakat domain. Specifically, the researchers aim to improve the performance of Indonesian NER in identifying and extracting named entities related to zakat. This is

important because NER is a crucial step in the information extraction process. Accurate identification and extraction of named entities can facilitate knowledge acquisition about zakat and improve understanding of this important obligation among Muslims in Indonesia. By utilizing the Conditional Random Fields (CRF) method and defining named entity classes for the zakat domain, the researchers hope to improve Indonesian NER in this domain and address the problem of low accuracy and efficiency.

CRFs are particularly well-suited for NER because they can account for contextual information and dependencies between the words in a sentence, which can be important for accurately identifying named entities. CRFs work by modeling the conditional probability of each word in a sentence given the previous words in the sentence and a set of labels (e.g., named entity types). This allows the CRF to consider the context and dependencies between the words in the sentence, rather than just considering each word in isolation. Additionally, CRFs can be trained using a large dataset of annotated text, where the named entities have been manually labelled. This allows the CRF to learn the patterns and features indicative of different named entities and make more accurate predictions when applied to new data. Overall, CRFs are a useful tool for NER because they can consider contextual information and dependencies between words and can be trained on large datasets of annotated text.

The study found that the performance of the Indonesian-NER model was evaluated using the Conditional Random Fields (CRF) method, resulting in an average precision of 0.902, recall of 0.834, and F1-score of 0.867. The F1-score is above 80%, indicating a high level of accuracy, and this result shows improvement from similar research conducted previously.

ACKNOWLEDGEMENTS

Research Center UIN Syarif Hidayatullah Jakarta Grant supported this research. We thank our colleagues from BAZNAS who provided insight and expertise that greatly assisted the research.

REFERENCES

[1] D. S. Rachmad, "Review Named Entity Recognition dengan Menggunakan Machine Learning," *J. Sains dan Inform.*, vol. 6, no. 1, pp. 28–33, 2020, doi: 10.34128/jsi.v6i1.204.
[2] H. L. Chieu and H. T. Ng, "Named entity recognition with a maximum entropy approach," pp. 160–163, 2003, doi: 10.3115/1119176.1119199.
[3] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural Architectures for Named Entity Recognition," *Proc. 2016 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol.*, pp. 260–270, 2016, doi: 10.18653/v1/N16-1030.
[4] H. T. Sukmana, J. M. Muslimin, A. F. Firmansyah, and L. K. Oh, "Building the Knowledge Graph for Zakat (KGZ) in Indonesian Language," *ASM Sci. J.*, vol. 16, pp. 1–10, 2021, doi: 10.32802/asmscj.2021.758.
[5] D. Khairani, D. A. Bangkit, N. F. Rozi, S. U. Masruroh, S. Oktaviana, and T. Rosyadi, "Named-Entity Recognition and Optical Character Recognition for Detecting Halal Food Ingredients: Indonesian Case Study," pp. 01–05, Nov. 2022, doi: 10.1109/CITSM56380.2022.9935966.
[6] Y. Wibisono and M. L. Khodra, "Pengenalan Entitas Bernama Otomatis untuk Bahasa Indonesia dengan Pendekatan Pembelajaran Mesin," 2018, doi: 10.31227/osf.io/vud2p.
[7] S. Morwal, "Named Entity Recognition using Hidden Markov Model (HMM)," *Int. J. Nat. Lang. Comput.*, 2012, doi: 10.5121/ijnlc.2012.1402.
[8] G. Paliouras, V. Karkaletsis, G. Petasis, and C. D. Spyropoulos, "Learning Decision Trees for Named-Entity Recognition and Classification," in *ECAI Workshop on Machine Learning for Information Extraction*, 2000.
[9] F. Riaz, M. W. Anwar, and H. Muqades, "Maximum Entropy based Urdu Named Entity Recognition," 2020, doi: 10.1109/ICEET48479.2020.9048203.
[10] A. D. Putra and A. S. Girsang, "Analysis of named-entity effect on text classification of traffic accident data using machine learning," *Indones. J. Electr. Eng. Comput. Sci.*, 2022, doi: 10.11591/ijeecs.v25.i3.pp1672-1678.
[11] S. Song, N. Zhang, and H. Huang, "Named entity recognition based on conditional random fields," *Cluster Comput.*, 2019, doi: 10.1007/s10586-017-1146-3.
[12] A. S. Wibawa and A. Purwarianti, "Indonesian Named-entity Recognition for 15 Classes Using Ensemble Supervised Learning," *Procedia Comput. Sci.*, vol. 81, no. May, pp. 221–228, 2016, doi: 10.1016/j.procs.2016.04.053.
[13] J. Li, A. Sun, J. Han, and C. Li, "A Survey on Deep Learning for Named Entity Recognition," *IEEE Trans. Knowl. Data Eng.*, pp. 1–1, 2020, doi: 10.1109/tkde.2020.2981314.
[14] B. Aryoyudanta, T. B. Adji, and I. Hidayah, "Semi-supervised learning approach for Indonesian Named Entity Recognition (NER) using co-training algorithm," *Proceeding - 2016 Int. Semin. Intell. Technol. Its Appl. ISITIA 2016 Recent Trends Intell. Comput. Technol. Sustain. Energy*, pp. 7–12, 2017, doi: 10.1109/ISITIA.2016.7828624.
[15] W. Gunawan, D. Suhartono, F. Purnomo, and A. Ongko, "Named-Entity Recognition for Indonesian Language using Bidirectional LSTM-CNNs," *Procedia Comput. Sci.*, vol. 135, pp. 425–432, 2018, doi: 10.1016/j.procs.2018.08.193.
[16] U. Wahyudin, "Sosialisasi Zakat untuk Menciptakan Kesadaran Berzakat Umat Islam," *J. Masy. Dan Filantr. Islam*, vol. 1, no. 1, pp. 17–20, 2018.
[17] D. D. A. Yani, H. S. Pratiwi, and H. Muhardi, "Implementasi Web Scraping untuk Pengambilan Data pada Situs Marketplace," *J. Sist. dan Teknol. Inf.*, vol. 7, no. 4, p. 257, 2019, doi: 10.26418/justin.v7i4.30930.
[18] D. A. A. D. K. Siti Ummi Masruroh, "Penerapan Algoritma Paice atau Husk untuk Stemming pada Kamus Bahasa Inggris ke

Bahasa Indonesia," *J. Tek. Inform.*, 2013, doi: 10.15408/jti.v6i2.2031.
[19] M. Irfan and A. F. Hidayatullah, "Tinjauan Literatur Named Entity Recognition dengan Machine Learning dan Deep Learning pada Ulasan Wisata," 2019.
[20] Rabi Narayan Behera, "A Survey on Machine Learning: Concept, Algorithms and Applications," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 5, no. 2, pp. 8198–8205, 2017.
[21] M. Batta, "Machine Learning Algorithms - A Review," *Int. J. Sci. Res. (IJ*, vol. 9, no. 1, pp. 381–386, 2020, doi: 10.21275/ART20203995.