International Journal of Electrical and Computer Engineering (IJECE)
Vol. 14, No. 5, October 2024, pp. 5489~5501
ISSN: 2088-8708, DOI: 10.11591/ijece.v14i5.pp5489-5501

Named entity recognition on Indonesian legal documents: a dataset and study using transformer-based models

Evi Yulianti[1], Naradhipa Bhary[1], Jafar Abdurrohman[1], Fariz Wahyuzan Dwitilas[1], Eka Qadri Nuranti[2], Husna Sarirah Husin[3]
[1]Department of Computer Science, Faculty of Computer Science, Universitas Indonesia, Depok, Indonesia
[2]Faculty of Computer Science, Institut Teknologi Bacharuddin Jusuf Habibie, Parepare, Indonesia
[3]School of Computer Science, Faculty of Innovation and Technology, Taylor's University, Selangor, Malaysia

Article Info

Article history:

Received Jan 22, 2024
Revised Jun 21, 2024
Accepted Jul 2, 2024

Keywords:

IndoBERT
M-BERT
Named entity recognition legal documents transformer
RoBERTa
XLM-RoBERTa

ABSTRACT

The large volume of court decision documents in Indonesia poses a challenge for researchers to assist legal practitioners in extracting useful information from the documents. This information can also benefit the general public by improving legal transparency, law enforcement, and people's understanding of the law implementation in Indonesia. A natural language processing task that extracts important information from a document is called named entity recognition (NER). In this study, the NER task is applied to legal domains, which is then referred to as legal entity recognition (LER) task. In this task, some important legal entities, such as judges, prosecutors, and advocates, are extracted from the decision documents. A new Indonesian LER dataset is built, called IndoLER data, consisting of approximately 1K decision documents with 20 types of fine-grained legal entities. Then, the transformer-based models, such as multilingual bidirectional encoder representations from transformers (BERT) or M-BERT, Indonesian BERT or IndoBERT, Indonesian robustly optimized BERT pretraining approach (RoBERTa) or IndoRoBERTa, XLM (cross lingual language model)-RoBERTa or XLMR, are proposed to solve the Indonesian LER task using this dataset. Our experimental results show that the RoBERTa-based models, such as XLM-R and IndoRoBERTa, can outperform the state-of-the-art deep-learning baselines using BiLSTM (bidirectional long short-term memory) and BiLSTM-conditional random field (BiLSTM-CRF) approaches by 7.2% to 7.9% and 2.1% to 2.6%, respectively. XLM-RoBERTa is shown to be the best-performing model, achieving the F1-score of 0.9295.

Corresponding Author:

Evi Yulianti
Department of Computer Science, Faculty of Computer Science, Universitas Indonesia
Depok, Indonesia
Email: evi.y@cs.ui.ac.id

1. INTRODUCTION

In Indonesia, a decision document is a statement made by a judge during an open court session that is documented to be made available and accessible to the public. It consists of some information about a particular legal case, such as legal facts, indictments, and punishment. As a legal state, Indonesia has an active judicial system, with approximately 100,000 decision documents produced by the judicial institutions every month [1]. The large volume of these documents poses a challenge for legal professionals in Indonesia to capture the knowledge contained in the documents more effectively and efficiently. This further opens the avenues for researchers to assist legal practitioners in extracting important information from the decision documents. In essence, this information can benefit not only legal practitioners, but also the general public by improving legal transparency, law enforcement, and people's understanding of the law implementation in Indonesia.

A natural language processing (NLP) task that extracts important information from a document is called named entity recognition (NER). NER plays an important role in extracting structured information from unstructured data. NER can be applied to various problem domains, such as social media [2], medical [3], biomedical [4], clinical [5], agriculture [6], and legal [1], [7] data. In general, NER works by identifying words in a document that represent important information such as persons, locations, dates, organizations, and events. These words are called entities. The entity results from NER can be used in further NLP tasks, such as location prediction [8], document summarization [9], and information retrieval [10].

In this study, the NER task is applied to legal domain, which is then referred to as legal entity recognition (LER) task. Some legal entities are extracted from decision documents in this task, such as judges, prosecutors, and advocates. In practice, these extracted legal entities are useful for some real use cases, such as field retrieval, i.e., to retrieve some decision documents that are related to a particular field/entity. For example, to find references about previous judges' decisions on theft cases, a legal practitioner can search for decision documents that contain the term "theft" in the *case type* entity. This is possible because initially, the LER task has identified all legal entities contained in each decision document.

Relatively a few studies have investigated the LER task in Indonesian-language dataset. Solihin and Budi [7] performed the LER task using a traditional rule-based method and 150 decision documents (with 11 types of legal entities). Later, Nuranti and Yulianti [1] conducted the LER task using more advanced deep-learning models, such as convolutional neural network (CNN), long short term memory (LSTM), bidirectional LSTM (BiLSTM), the combination of these models with conditional random fields (CRF) method, and 1,000 decision documents (with 10 types of legal entities). All this research, however, did not make their dataset publicly available, making it difficult to reproduce the results. Their dataset also employs a smaller number of legal entities compared to ours which consists of 20 types of fine-grained legal entities. At last, they still did not investigate the use of state-of-the-art transformer-based models for solving the Indonesian LER task. All of these aspects differentiate between our work and their work, which highlights the contributions of this study. To sum up, our contributions in this work are as follows:

- We create a new publicly available dataset on the Indonesian legal entity recognition task, called IndoLER (*http://github.com/ir-nlp-csui/indoler*) dataset, consisting of approximately 1K decision documents and 20 types of fine-grained legal entities (±6M words and ±25K annotated entities), that are manually annotated by human annotators.
- We propose using various state-of-the-art transformed-based models, such as multilingual bidirectional encoder representations from transformers (BERT) or M-BERT [11], Indonesian BERT or IndoBERT [12], Indonesian robustly optimized BERT pretraining approach (RoBERTa) or IndoRoBERTa [13], XLM (cross lingual language model)-RoBERTa or XLM-R [14] for solving the Indonesian LER task. Note that none of the previous work has studied the effectiveness of a transformer-based model for extracting legal entities from Indonesian decision documents.

2. RELATED WORK

Aletras *et al.* [15] predicted the decisions for a particular case using the models learned from legal documents. They built a binary classification model using English court decision documents and support vector machine (SVM) algorithm. The results show that the model achieves an accuracy of 79%. Medvedeva *et al.* [16] used the same dataset and classification algorithm as [15], but they only focused on the procedure and fact sections of the documents. Kowsrihawat *et al.* [17] used decision documents written in the Thai language from the Thailand Supreme Court. Their models used naive Bayes, SVM, bidirectional gated recurrent unit (Bi-GRU), and Bi-GRU with attention to predict the decision. The results show that the Bi-GRU with attention model performs the best with a macro F1-score of 63.34%. Next, Virtucio *et al.* [18] used English decision documents from the Philippine Supreme Court to build a decision classification model. Their results using linear SVM yielded the best accuracy of 45% on n-grams datasets and 55% on topic datasets.

Different from prior studies that predicted the decision for a legal case, the research by Nuranti and Yulianti [19] aims to predict the category and duration of punishment from Indonesian decision documents. The punishment categories are divided into four types: mild, moderate, heavy, and very heavy. Their research used some powerful deep-learning models, such as BiLSTM, CNN, LSTM, and the combination of these models with attention. For both tasks of predicting punishment categories and length, the *CNN+Attention* model was shown to achieve the best performance.

Barriere and Fouret [20] studied the extraction of general entities from the decision documents. They performed named entity recognition (NER) task using French decision documents to extract four general entities: person, court members, location, and date. The dataset consists of 94 manually annotated original court decision documents by legal experts. In contrast, other research has paid more attention to the legal-specific entities as the results of the NER task, such as judges, prosecutors, and punishment. Leitner *et al.* [21] conducted the NER task using German court decision documents and the BiLSTM-CRF approach to extract 19 fine-grained entities and 7 coarse-grained entities. Their dataset consists of 750 German court decision documents published through the *Rechtsprechung im* internet online portal. Next, Correia *et al.* [22] studied the NER task in the Portuguese language using Brazilian court decision documents and BiLSTM-CRF approach. Four coarse-grained entities and 21 fine-grained entities were extracted in their study. Their dataset consists of 594 Brazilian court decision documents annotated by 76 law students.

Studies on NER in Indonesian decision documents are still relatively rare. Nuranti and Yulianti [1] identified 10 legal entities using machine learning and deep-learning models with 1,000 Indonesian decision documents. Their methods include SVM, CNN, LSTM, BiLSTM, and the combination of these models with CRF method. Solihin and Budi [7] detected 11 legal entities using a rule-based approach with only 150 Indonesian decision documents. Different from both researchers, who did not publish their dataset, we make our dataset publicly available in this work. Our dataset contains gold standard data annotated by humans as opposed to automatic annotation as in [1], and it contains a larger number of documents and more fine-grained entities (consisting of 1,000 documents and 20 types of legal-specific entities) compared to [7]. Next, In contrast to [1] and [7] who used rule-based, traditional machine learning, and deep learning methods, this work investigates some state-of-the-art transformer-based models to perform the NER task on Indonesian legal documents, referred to as Indonesian legal entity recognition task.

3. DATASET CREATION

3.1. Data collection

The data used in this study consists of 1,000 court decision documents in criminal cases adjudicated by the district court. These documents are obtained from the list of document IDs used in [1]. From the given list of document IDs, the authors scraped the original decision documents from the Supreme Court's online portal [23]. The output of this stage is a collection of PDF files of Indonesian decision documents. The authors obtained 993 successfully downloaded PDF files. The remaining seven PDF files could not be downloaded because they were verdicts against minors which were not published or on restricted access.

3.2. Data extraction and cleaning

Before proceeding with data processing, text information is initially extracted from the PDF files. Because the extracted text still contains a lot of noise, the data-cleaning process is performed. This process includes removing the Supreme Court's header and footer as well as page numbers, merging separated letters with spaces, deleting empty lines, removing excessive spaces, removing excessive symbol repetitions, formatting numbers, and removing non-ASCII characters.

3.3. Data annotation

After the data-cleaning process is performed, the next step is to perform manual data annotation by annotators using 20 types of legal entities presented in Table 1. These legal entities are obtained from the analysis of legal entities used in previous studies on Indonesian LER task [1], [7] as described in Table 2, with the addition of four new entities. There are a few entities that have been annotated in the previous work, but in more general entities. For example, Nuranti and Yulianti [1] identified the entity "*nama terdakwa*" (defendant's name) and "*nama saksi*" (witness name) as the same entity "*terlibat*" (person). However, in this study, they are separated into two different entities to make more fine-grained entities. As a result, users can distinguish the defendants and witnesses in the results of extracted entities. Four new entities are determined in this study after performing a careful analysis of the content of the decision documents: "*tingkat kasus*" (case level), "*jenis perkara*" (case type), "*jenis dakwaan*" (indictment type), and "*jenis amar*" (decision type).

Nakayama [24] use Doccano as the annotation tool to make the annotators easier in conducting the annotation process. The annotation guideline is also created before the annotation is conducted to help the annotators better understand the task as well as the procedure to annotate the documents using Doccano. A pilot test is also conducted using five document samples in order to familiarize the annotators with the annotation task. To ensure the quality of the annotation, the results of the annotation process are also checked by a researcher in this study who is already very familiar with the annotation of legal documents.

Table 1. List of legal entities in our dataset

No	Legal entity	Description
1	*Nomor putusan* (Decision call number)	Case registration number
2	*Nama pengadilan* (Court)	Name of the first-level court authorized to adjudicate cases based on its jurisdiction
3	*Tingkat kasus* (Case level)	Level of case examination by the court based on the hierarchy regulated by the law, consisting of "*pemeriksaan tingkat pertama*" (first level), "*banding*" (appeal), "*kasasi*" (cassation), and "*peninjauan kembali*" (judicial review).
4	*Nama terdakwa* (Defendant)	Name of the person accused and examined in the trial
5	*Jenis perkara* (Case type)	Criminal offense charged by the prosecutor against the defendant based on applicable laws and regulations.
6	*Pasal tuntutan* (Lawsuit article)	Legal basis of the charges brought by the prosecutor against the defendant.
7	*Tuntutan hukuman* (Lawsuit)	Type of punishment and duration of punishment claimed by the prosecutor
8	*Nama saksi* (Witness)	Name of the person who can provide testimony for the purposes of investigation, prosecution, and trial based on what they have heard, seen, or experienced
9	*Tanggal kejadian* (Case date)	Date of the occurrence of the act as legal facts in the trial based on the evidence.
10	*Jenis dakwaan* (Indictment type)	Type of indictment prepared by the public prosecutor based on the likelihood of proving the indictment, consisting of "*dakwaan tunggal*" (single indictment), "*dakwaan alternatif*" (alternative indictment), "*dakwaan subsidair*" (subsidiary indictment), "*dakwaan kumulatif*" (cumulative indictment), and "*dakwaan kombinasi*" (combined indictment)
11	*Pasal dakwaan* (Indictment Article)	Legal basis of the indictment prepared by the prosecutor against the defendant for the alleged criminal offense.
12	*Pasal pertimbangan hukum* (Law consideration article)	Legal basis considered by the judge to render a verdict after considering the facts presented in the trial.
13	*Amar putusan* (Verdict)	Type of verdict rendered by the judge in criminal cases, namely acquittal, release, or conviction
14	*Putusan hukuman* (Punishment)	Type of punishment and duration or amount of punishment imposed by the judge in a legally binding verdict.
15	*Tanggal putusan* (Decision date)	Date when the conclusion of a case is decided by the judges in the Council of Judges' Deliberation
16	*Nama hakim ketua* (Judge)	Name of the judge who presides over the trial proceedings.
17	*Nama Hakim Anggota* (Member judge)	Names of the other judges who examine, adjudicate, and decide cases together with the presiding judge.
18	*Nama panitera* (Registrar)	Name of the court officer responsible for the administration of the trial.
19	*Nama pengacara* (Advocate)	Name of the legal expert authorized to represent parties involved in legal proceedings.
20	*Nama jaksa* (Prosecutor)	Name of the functional officer who acts as a public prosecutor and implements legally binding court decisions.

Table 2. The comparison between our legal entities and the legal entities used in previous work

No	Legal Entity	Nuranti and Yulianti [1]	Solihin and Budi [7]
1	*Nomor Putusan* (Decision call number)	✓	✓
2	*Nama Pengadilan* (Court)	O (organization)	×
3	*Tingkat Kasus* (Case level)	×	×
4	*Nama Terdakwa* (Defendant)	O (person)	✓
5	*Jenis Perkara* (Case type)	×	×
6	*Pasal Tuntutan* (Lawsuit article)	O (law)	×
7	*Tuntutan Hukuman* (Lawsuit)	×	✓
8	*Nama Saksi* (Witness)	O (person)	×
9	*Tanggal Kejadian* (Case date)	O (date)	×
10	*Jenis Dakwaan* (Indictment type)	×	×
11	*Pasal Dakwaan* (Indictment article)	O (law)	✓
12	*Pasal Pertimbangan Hukum* (Law consideration article)	O (law)	×
13	*Amar Putusan* (Verdict)	×	×
14	*Putusan Hukuman* (Punishment)	✓	✓
15	*Tanggal Putusan* (Decision Date)	O (date)	✓
16	*Nama Hakim Ketua* (Judge)	O (person)	✓
17	*Nama Hakim Anggota* (Member Judge)	O (person)	✓
18	*Nama Panitera* (Registrar)	✓	✓
19	*Nama Pengacara* (Advocate)	✓	×
20	*Nama Jaksa* (Prosecutor)	✓	✓

Note: ✓: The legal entity is used in the study. ×: The legal entity is not used in the study. O: The legal entity is used in a more general entity.

The annotation process is carried out by three bachelor students in Computer Science using 20 fine-grained legal entities that are described earlier in Table 1. The annotation is conducted in two stages: i) annotating overlapping documents and ii) annotating unique documents. In the first stage, the annotators annotate the same documents because we want to compute the agreement scores between annotators. The final labels are then determined using the majority vote technique. This stage uses 10% of the data (i.e., 10 documents), and the agreement scores are computed using Fleiss' Kappa. Any discrepancies in the annotation results are evaluated to improve the quality of the annotation in the second stage. Based on the annotation results of the overlapping documents, the authors calculated the value of inter-annotator agreement (IAA) using Fleiss's Kappa method to compute the agreement between annotators. The resulting IAA value is 0.9519, which indicates a very strong agreement according to [25]. It means that the annotators produced highly similar annotation results because their annotations strongly agree with each other.

The second stage of annotation is then performed on unique documents for each annotator. Here, each annotator uniquely annotates 30% of the data (328 docs). The output of the annotation process from Doccano is the text that has been labeled in the begin inside outside (BIO) format, which is the common format used in NER task. This is our final dataset that is created in this study, named IndoLER (*http://github.com/ir-nlp-csui/indoler*) dataset, and it has been made publicly available for research purposes.

3.4. Data analysis

The authors conduct an analysis of the overall statistics of the distribution of legal entities in our dataset. The overall statistics of our dataset are presented in Table 3. We can see that our dataset consists of approximately 1K documents containing ±6M words and ±25K annotated entities.

Table 3. The statistics of our dataset

Item	Value
#decision documents	993
#words	5,894,080
Avg count of words per document	5,935.63
#annotated entities	24,845
#tagged words (“B” and “I” tags)	124,395
#untagged words (“O” tag)	5,769,685

Table 4 describes the number of annotated entities for each legal entity, the number of words tagged as legal entity, and the average number of words tagged as legal entity. An entity that appears most frequently in our dataset is “*nama saksi*” (witness) because a legal case can have several witnesses. On the other hand, the entity that appears the least is “*nama pengacara*” (advocate) because not all legal case has an advocate. We found that only 18% of documents in our dataset contain advocate names. In other words, most of the legal cases are not handled by the advocates.

Table 4. The statistics of legal entities in our dataset

No	Legal Entity	Count	#words tagged as entity	Avg #words tagged as entity
1	*Nomor putusan* (Decision call number)	993	1,809	1.82
2	*Nama pengadilan* (Court)	993	3,248	3.27
3	*Tingkat kasus* (Case level)	975	2,097	2.15
4	*Nama terdakwa* (Defendant)	1,236	5,418	4.38
5	*Jenis perkara* (Case type)	950	9,350	9.84
6	*Pasal tuntutan* (Lawsuit article)	951	9,418	9.90
7	*Tuntutan hukuman* (Lawsuit)	1,201	9,863	8.21
8	*Nama saksi* (Witness)	3,948	12,394	3.14
9	*Tanggal kejadian* (Case date)	1,635	4,863	2.97
10	*Jenis dakwaan* (Indictment type)	869	2,702	3.11
11	*Pasal dakwaan* (Indictment article)	1,368	14,166	10.35
12	*Pasal pertimbangan hukum* (Law consideration article)	1,332	13,044	9.79
13	*Amar putusan* (Verdict)	989	4,243	4.29
14	*Putusan hukuman* (Punishment)	1,173	11,617	9.9
15	*Tanggal putusan* (Decision date)	987	2,987	3.03
16	*Nama hakim ketua* (Judge)	990	3,511	3.55
17	*Nama hakim anggota* (Member judge)	1,944	6,887	3.54
18	*Nama panitera* (Registrar)	990	2,649	2.68
19	*Nama pengacara* (Advocate)	342	1,102	3.22
20	*Nama jaksa* (Prosecutor)	979	3,027	3.09

Next, the entity that is shown to have the longest words is "*pasal dakwaan*" (indictment article), followed by the "*pasal pertimbangan hukum*" (law consideration article). It is understandable that indictment and law consideration article entities have lengthy text because they describe the list of articles that are associated with a particular legal case. The entity "*nama saksi*" (witness name) is also shown to contain a lot of words because as explained earlier, there are usually a lot of witnesses for a particular legal case. Then, the entity that contains the shortest text is "*nomor putusan*" (decision number).

3.5. Data splitting

For our experiment, our annotated datasets are then divided into training and testing data. The proportion of training data used is 70% of the dataset (695 docs). Of the remaining 30% of the data, 90% is used as the testing data (268 docs) and 10% is used as the validation data (30 docs). The data separation is done using the stratified shuffle split method described by Sechidis *et al.* [26]. This method is used to ensure that the distribution of each entity class is preserved in each part of the data.

4. TRANSFORMER-BASED MODEL FOR LEGAL ENTITY RECOGNITION

In general, according to the language of the corpus that is used to pre-train the models, the transformer-based models can be divided into two groups: multilingual and monolingual language models. A multilingual language model is a language model that utilizes multiple languages (more than one) in the pre-trained dataset. On the other hand, a monolingual language model is a language model that uses a single language in the pre-trained dataset, e.g., Indonesian.

This work uses two multilingual language models (M-BERT and XLM-RoBERTa), and two monolingual language models (IndoBERT and Indonesian RoBERTa). These models are chosen because they have been shown to perform well in various NLP tasks [12]–[14], [27]. None of the previous work, however, has investigated the use of transformer-based models for the named entity recognition on legal documents, i.e., legal entity recognition. In this study, these models are fine-tuned using our dataset to solve our specific task on Indonesian legal entity recognition. The fine-tuning process is performed by adding an additional output layer in the architecture of each of these systems and retraining the model using our dataset to adjust the optimal weights of model parameters obtained previously in the pre-training process.

The flow of process in this work is illustrated in Figure 1. In general, the dataset creation steps including data extraction, cleaning, and annotation processes are performed first to produce the ground truth labels in our dataset. Next, tokenization is performed on our dataset using a specific tokenizer for each model. The fine-tuning process is then performed to fine-tune the pre-trained transformer-based models for Indonesian legal entity recognition (LER) task using our annotated dataset. The resulting models are our LER systems to be used in the experiments in this study. The explanation of each of the transformer-based models used in this study is detailed in the following subsections.

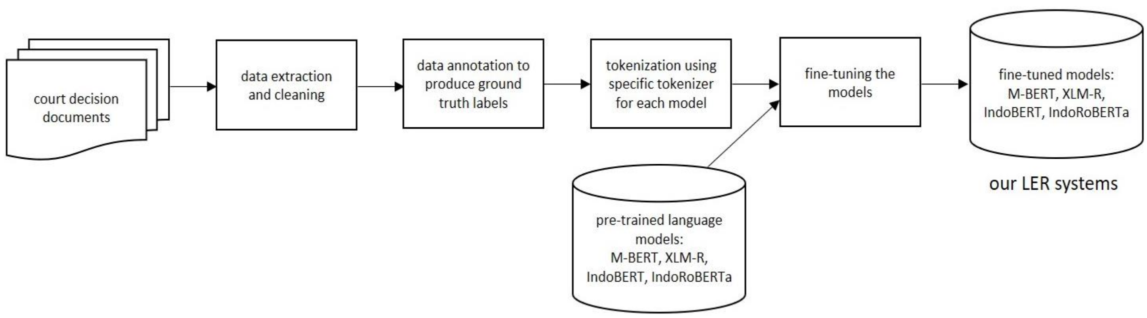

Figure 1. The flow of process to develop our LER systems

4.1. Multilingual language model

4.1.1. Multilingual BERT (M-BERT)

M-BERT [11] is a variant of the original BERT model by Devlin *et al.* [27] that was pre-trained using multilingual data sources. M-BERT was pre-trained using Wikipedia articles in 104 languages, including the Indonesian language. Due to the varying data sizes across languages, the data per language was scaled. Similar to the original BERT model, the M-BERT model was also pre-trained using the masked language modelling (MLM) task which trains the model to fill in the blank parts of the text, as well as the next sentence prediction (NSP) task, which trains the model to predict whether the next sentence is the continuation of a given sentence. In terms of architecture, the M-BERT model has the same configuration as the original BERT model. It is composed of a stack of Transformer encoders that will encode the representation of a language. M-BERT uses a hidden size of 768, an intermediate size of 3072, and a position embedding of 512. The number of hidden layers and attention heads used is also the same, which is 12.

4.1.2. XLM-RoBERTa (XLM-R)

XLM-RoBERTa or XLM-R is an extension of the RoBERTa model developed in [14]. XLM-R was pre-trained using a dataset which is based on CommonCrawl data in 100 languages. The dataset is over 2 terabytes in size. Similar to [27], [28], this model was pre-trained using the masked language modeling task. XLM-R has two variants, namely base and large. These variants differ from each other in terms of model parameters. XLM-R_{base} has 12 layers, 768 hidden units, and 12 attention heads, with a total of 270 million parameters. On the other hand, XLM-R_{large} has 14 layers, 1024 hidden units, and 16 attention heads, with a total of 550 million parameters.

4.2. Monolingual language model

4.2.1. IndoBERT

IndoBERT is a BERT-based model that was pre-trained by Wilie *et al.* [12] using a large size of Indonesian dataset. In the original paper, the effectiveness of IndoBERT was also tested to solve a range of NLP tasks. However, the performance of IndoBERT for legal entity recognition tasks has not been examined. IndoBERT was pre-trained using the Indo4B dataset, an Indonesian language dataset containing four billion words or approximately 250 million sentences, with a dataset size of approximately 23 GB. The Indo4B dataset is compiled using 15 datasets that contain formal, informal language (everyday language), and code-mixed languages. IndoBERT has two variants, namely base and large, with each variant respectively has 124.5 and 335.2 million parameters. The IndoBERT pre-training was conducted in two phases. In the first phase, the model was trained using a maximum sequence length of 128, while in the second phase, a maximum sequence length of 512 was used.

4.2.2. Indonesian RoBERTa (IndoRoBERTa)

Indonesian RoBERTa or IndoRoBERTa is a RoBERTa model [13] that was pre-trained using an Indonesian dataset. IndoRoBERTa was developed by pre-training the RoBERTa architecture using the OSCAR dataset [29] written in the Indonesian language with the size of 17.05 GB. It utilizes the $RoBERTa_{base}$ architecture with a total of 124 million parameters.

5. EXPERIMENT

5.1. Fine-tuning process of our transformer-based models

The only pre-processing step performed before fine-tuning the models is tokenization using the tokenizer of each model. As described in the original paper [27], BERT models do not require pre-processing steps such as removing punctuation marks and stop words, as it can decrease the accuracy of the model. After the tokenization is performed, the fine-tuning process is performed for each of the M-BERT, XLM-R, IndoBERT, and IndoRoBERTa models.

Our experiment uses both variants of the XLM-R pre-trained models, namely XLM-R_{base} and XLM-R_{large}, following the original paper of XLM-R in [14]. Then, we use BERT-base-multilingual-uncased variant of M-BERT because the data created by the author has been lowercased following the research in [1]. For IndoBERT, we use all variants provided from the original paper [12], namely $IndoBERT_{base}$ and $IndoBERT_{large}$. As for Indonesian RoBERTa, the author used the base version provided in [29]. The hyperparameter settings for our transformer-based models in the experiment are listed in Table 5. As can be seen from the table, various values for epoch, sequence length, learning rate, and stride parameters are tested in our experiment.

Table 5. The variation of hyperparameter values of our models tested in our experiment

Hyperparameter	Value
Epoch	2, 4, 8, 16
Batch size	8
Sequence length	128, 256, 52
Learning rate	2e-5, 3e-5, 5e-5
Weight decay	0.01
Stride	0%, 10%, 25%, 50%

5.2. Computing environment

The authors implemented the code using Python programming language version 3.8. The development and execution of the program for the training and inference stages are performed on the DGX A100 computer environment owned by Faculty of Computer Science, Universitas Indonesia. The specifications of the computing environment can be seen in Table 6.

Table 6. System specification of the development environment

Component	Specification	Allocated space
Processor	AMD EPYC™ 7742 (64 Core)	Unlimited
RAM	1056.65 GB	64 GB
GPU	8x NVIDIA A100 (40 GB VRAM)	1 GPU

5.3. Baseline systems

The baseline systems for this study are the two best-performing methods in previous work on Indonesian LER task [1]. They include the bidirectional long short-term memory (BiLSTM) and the combination of BiLSTM with the conditional random fields (BiLSTM-CRF) models. BiLSTM is a variant of the long short-term memory (LSTM) model that stores and processes input in forward and backward directions [30]. This allows BiLSTM to process contextual information more effectively, and therefore, it is commonly used in tasks that require context in their processing, such as named entity recognition. In the BiLSTM-CRF model, BiLSTM is used to capture contextual information from the text, while CRF is used to calculate tag transfer scores and determine the most accurate tag sequence for entity annotation [31]. The implementation of the CRF model together with the BiLSTM model has been proven effective in improving the accuracy of entity recognition and other NLP tasks [1], [32].

For the BiLSTM and BiLSTM-CRF models, the authors follow the pre-processing steps in [1]. The input text data is converted to lowercase, numbers in the text are changed to “$<$XXX$>$” with the frequency of “X” corresponds to the frequency of digits, all punctuation marks in the text are removed, and finally, the text is tokenized per word. The hyperparameter settings for these models follow the variation described in Table 5 earlier, except for the learning rate. The learning rate value for these models is 0.001, following the value used in [1]. Then, the word embedding dimensions employed for these models are 100, 200, and 500.

5.4. Evaluation metrics

All models are evaluated by comparing the predicted results with the ground truth labels in our dataset. The evaluation metrics used are precision, recall, and F1-score. Precision and recall are commonly used to measure the performance of models in tasks related to information extraction, such as NER. Then F1-score is also a common metric for NER task, which takes a balance between precision and recall.

6. RESULTS AND ANALYSIS

6.1. The results of overall model performance

The performance comparison of all models in this work can be seen in Table 7. Based on our experimental results, we found that the XLM-R and IndoRoBERTa models gain the best performance, with the highest F1-score of 0.9292 achieved by the XLM-R$_{\text{large}}$ model. The XLM-R$_{\text{large}}$ model outperforms all the baseline models by 7.67% over the BiLSTM method and by 2.64% over the BiLSTM-CRF method. This indicates the superior performance of the XLM-R and IndoRoBERTa models to solve the Indonesian LER task. The XLM-R$_{\text{base}}$ version achieves a slightly lower score compared to the large version, gaining an F1-score of 0.9281. It obtains 7.71% and 2.48% increases compared to the BiLSTM and BiLSTM-CRF baselines, respectively. Besides XLM-R models, IndoRoBERTa can also improve all the baseline models by achieving an F1-score of 0.9246.

The M-BERT and IndoBERT models demonstrate fair performance. The M-BERT model still shows a slight increase over the BiLSTM baseline, but it is still worse than the BiLSTM-CRF baseline. The IndoBERT models, however, are shown to be the least accurate among all of our models, as they cannot lead to any improvement over both baseline models. The poor performance of IndoBERT could be due to the dataset used for pre-training IndoBERT that contains a significant amount of non-formal Indonesian language (colloquial) sourced from Twitter. In addition, it also includes code-mixed data that contain a mixture of formal and non-formal languages. However, the legal documents used in this work are court decision documents, which all use the formal Indonesian language. As a result, the non-formal Indonesian language in the pretraining dataset of IndoBERT may not be useful for improving the accuracy of the NER model for Indonesian legal documents.

Table 7. The performance comparison of all models

Type	Model	Precision	Recall	F-1
Our models	$XLMR_{large}$	0.9124	0.9472	0.9295
	$XLMR_{base}$	0.9047	0.9528	0.9281
	M-BERT	0.8422	0.8422	0.8697
	$IndoBERT_{large}$	0.8454	0.8334	0.8394
	$IndoBERT_{base}$	0.8467	0.8263	0.8364
	IndoRoBERTa	0.9278	0.9215	0.9246
Baselines	BiLSTM	0.8873	0.8375	0.8617
	BiLSTM-CRF	0.9056	0.9071	0.9056

We then analyze the misclassification cases that often occur in each model. We divide the misclassifications into three groups: misclassifications into the tag "O", misclassifications between the begin (B) and inside (I) tags, and misclassifications into other entity classes. In Table 8, we can see that the majority of errors are misclassifications of entity into the tag "O". The author suspects that this occurs due to the imbalance of data between tagged and untagged words. In the dataset used, the ratio between untagged and tagged words is 46. Compared to the CoNLL 2003 dataset [33], which has a ratio of 4.9 between untagged and tagged words, it can be seen that this dataset is more biased towards the "O" tag.

Table 8. The misclassification errors of our models

Model	Tagged as O	Flipped between tags B and I	Other cases
$XLMR_{large}$	91.16%	4.04%	4.80%
$XLMR_{base}$	91.51%	4.76%	3.73%
M-BERT	89.53%	7.18%	3.29%
$IndoBERT_{large}$	79.30%	16.71%	3.99%
$IndoBERT_{base}$	83.24%	12.94%	3.82%
IndoRoBERTa	87.96%	7.23%	4.81%

6.2. The results for each legal entity

Table 9 presents the results of the best-performing model, XLM-R_{large}, to recognize each legal entity in testing data. The entity that has the highest F1-score is "*nomor putusan*" (decision number), which means that this entity can be recognized most accurately by the model. This can be explained because the entity "*nomor putusan*" has a unique format that starts with a number and contains some slash characters (e.g., 325/pid.b/2015/pn bwi) and mostly appear at the beginning of the document. Besides the entity "*nomor putusan*" (decision number), the entities "*hakim ketua*" (judge) and "*nama pengadilan*" (court name) can also be accurately predicted by the models. It is because they have clear patterns and structures in the documents. The author analyzes that the location of this entity is often at the beginning of the document and close to other entity classes, resulting in fewer "O" tokens found before and after the entity. This makes it easier for the model to understand the context surrounding the entity. Additionally, this entity is also almost always present in the documents.

Table 9. The performance of XLM-R_{large} model for each legal entity

No	Legal entity	F1-score
1	*Nomor putusan* (Decision call number)	0.9802
2	*Nama pengadilan* (Court)	0.9635
3	*Tingkat kasus* (Case level)	0.9593
4	*Nama terdakwa* (Defendant)	0.9616
5	*Jenis perkara* (Case type)	0.9403
6	*Pasal tuntutan* (Lawsuit article)	0.9143
7	*Tuntutan hukuman* (Lawsuit)	0.8853
8	*Nama saksi* (Witness)	0.9451
9	*Tanggal kejadian* (Case date)	0.6919
10	*Jenis dakwaan* (Indictment type)	0.8722
11	*Pasal dakwaan* (Indictment article)	0.8627
12	*Pasal pertimbangan hukum* (Law consideration article)	0.5939
13	*Amar putusan* (Verdict)	0.9368
14	*Putusan hukuman* (Punishment)	0.8682
15	*Tanggal putusan* (Decision date)	0.9618
16	*Nama hakim ketua* (Judge)	0.9692
17	*Nama hakim anggota* (Member judge)	0.9593
18	*Nama panitera* (Registrar)	0.9492
19	*Nama pengacara* (Advocate)	0.8631
20	*Nama jaksa* (Prosecutor)	0.9526

On the other hand, the entity class "*tanggal kejadian*" (case date) has the lowest F1-score, which means that it is the least accurate entity to be predicted by the model. It can be explained because this entity is rarely found in documents and does not follow the same occurrence patterns. Furthermore, the position of the "*tanggal kejadian*" entity is often in the middle of the document (in the legal facts section) and not close to other entity class tokens, resulting in many "O" tokens found before and after the entity. This makes it challenging for the model to understand the context surrounding the entity.

6.3. The results of optimal hyperparameter settings

Table 10 displays the best combination of hyperparameter values for our transformer-based models. For the monolingual language models IndoBERT and IndoRoBERTa, the best hyperparameters are as follows: the sequence length of 128, the epoch of 16, the learning rate of 2e-5, and the stride of 0. However, there is an exception for the best stride hyperparameter value for the $\text{IndoBERT}_{\text{base}}$ model, which is 32 (50%). For the multilingual language models XLM-R and M-BERT, the best hyperparameters are as follows: the sequence length of 512, the epoch of 16, the learning rate of 2e-5, and the stride of 0. Note that the best hyperparameter settings for the deep learning baseline models (BiLSTM and BiLSTM-CRF) are similar to those of $\text{IndoBERT}_{\text{large}}$ and IndoRoBERTa, with the most optimal word embedding dimension of 100.

Table 10. Best hyperparameter values for each model

Model	Epoch	Sequence length	Learning rate	Stride
$\text{XLMR}_{\text{large}}$	16	512	2e-5	0
$\text{XLMR}_{\text{base}}$	16	512	2e-5	0
M-BERT	16	512	2e-5	0
$\text{IndoBERT}_{\text{large}}$	16	128	2e-5	0
$\text{IndoBERT}_{\text{base}}$	16	128	2e-5	50%
IndoRoBERTa	16	128	2e-5	0

The authors found that the hyperparameter that consistently improves the performance across all tested models was only the epoch. Increasing the number of epochs improved the F1-score of the tested models. Overall, the author observed an improvement of 12.76% when increasing the epochs from two to four, 8.22% from four to eight, and 3.44% from eight to sixteen. Changing the values of other hyperparameters, however, has been shown to produce inconclusive results. In some cases, increasing these hyperparameter values improves the F1-score in some models, while in other cases, the F1-score is shown to decrease. For the sequence length hyperparameter, increasing its value consistently improves the performance of the multilingual model, but decreases the performance of the monolingual model. Regarding the stride hyperparameter, the changes in performance are inconsistent, sometimes it increases the performance, while in other cases it causes decreasing results. As for the learning rate hyperparameter, increasing its value does not improve the model's performance. This can be observed in both the monolingual and multilingual models, where a learning rate of 2e-5 performed better compared to values of 3e-5 and 5e-5.

7. CONCLUSION

This study investigates the NER task on Indonesian legal documents, which is referred to as LER task. A new dataset for the Indonesian LER task is created, consisting of approximately 1K decision documents with 20 types of fine-grained legal entities (±6M words and ±25K annotated legal entities). Besides developing a new benchmark for the LER task in the Indonesian language, this study also proposes some state-of-the-art transformer-based models, including multilingual and monolingual language models, to solve the LER task in the Indonesian language. The multilingual models include XLM-RoBERTa (XLM-R) and M-BERT, and the monolingual models include IndoBERT and Indonesian RoBERTa (IndoRoBERTa). Empirical evaluations are performed to examine the effectiveness of these models against state-of-the-art deep learning baselines: BiLSTM and BiLSTM-CRF.

According to our experimental results, the XLM-R and IndoRoBERTa show superior performance, outperforming all the baseline models in this study. The XLM-R (large version) model is shown to achieve the best performance among all models examined in this study. It demonstrates 7.9% and 2.64% improvement over the state-of-the-art baseline models BiLSTM and BiLSTM-CRF, respectively. The entities that gain the best prediction results are "*nomor putusan*" (decision number), "*nama hakim ketua*" (judge), and "*nama pengadilan*" (court); while those that are difficult to predict correctly are "*pasal pertimbangan hukum*" (law consideration article) and "*tanggal kejadian*" (case date).

In future, we plan to examine the use of legal entities extracted from decision documents for document retrieval tasks. This enables us to perform field retrieval in order to retrieve documents based on a particular field or entities. For example, we can retrieve documents that are related to a particular case type (e.g., theft). Further, it is also possible to explore the use of legal entity as features to predict the decision (e.g., guilty or not) or the punishment length of a particular case. At last, while in this study we perform the NER task on decision documents in criminal cases, further studies may be needed to perform the NER task on legal documents in civil cases to confirm the effectiveness of XLM-R model for this task. It is also potential to create a new NER dataset on civil cases decision documents and explore the differences between types of legal entities that could be extracted from civil cases and criminal cases decision documents.

ACKNOWLEDGEMENTS

This research was funded by the Directorate of Research and Development, Universitas Indonesia, under Hibah PUTI Q2 2022 (Grant No. NKB-571/UN2.RST/HKP.05.00/2022). The authors also wish to acknowledge Tokopedia-UI AI Center, Faculty of Computer Science Universitas Indonesia for providing the authors with some computational resources.

REFERENCES

[1] E. Q. Nuranti and E. Yulianti, “Legal entity recognition in Indonesian court decision documents using Bi-LSTM and CRF approaches,” *2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2020, doi: 10.1109/icacsis51025.2020.9263157.

[2] B. A. Benali, S. Mihi, I. El Bazi, and N. Laachfoubi, “New approach for Arabic named entity recognition on social media based on feature selection using genetic algorithm,” *International Journal of Electrical and Computer Engineering*, vol. 11, no. 2, pp. 1485–1497, Apr. 2021, doi: 10.11591/ijece.v11i2.pp1485-1497.

[3] I. J. Unanue, E. Z. Borzeshi, and M. Piccardi, “Recurrent neural networks with specialized word embeddings for health-domain named-entity recognition,” *Journal of Biomedical Informatics*, vol. 76, pp. 102–109, 2017, doi: 10.1016/j.jbi.2017.11.007.

[4] M. Bali, A. S. Pichandi, and J. H. Duraisamy, “Biomedical-named entity recognition using CUDA accelerated KNN algorithm,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 21, no. 4, pp. 825–835, Aug. 2023, doi: 10.12928/telkomnika.v21i4.24065.

[5] Ravikumar J. and R. Kumar P., “Machine learning model for clinical named entity recognition,” *International Journal of Electrical and Computer Engineering*, vol. 11, no. 2, pp. 1689–1696, Apr. 2021, doi: 10.11591/ijece.v11i2.pp1689-1696.

[6] Z. Liu, M. Luo, H. Yang, and X. Liu, “Named entity recognition for the horticultural domain,” *Journal of Physics: Conference Series*, vol. 1631, no. 1, Sep. 2020, doi: 10.1088/1742-6596/1631/1/012016.

[7] F. Solihin and I. Budi, “Recording of law enforcement based on court decision document using rule-based information extraction,” in *2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2018, pp. 349–354, doi: 10.1109/ICACSIS.2018.8618187.

[8] L. F. Simanjuntak, R. Mahendra, and E. Yulianti, “We know you are living in bali: Location prediction of twitter users using BERT language model,” *Big Data and Cognitive Computing*, vol. 6, no. 3, p. 77, 2022, doi: 10.3390/bdcc6030077.

[9] M. E. Khademi and M. Fakhredanesh, “Persian automatic text summarization based on named entity recognition,” *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, pp. 1–12, 2020, doi: 10.1007/s40998-020-00352-2.

[10] A. Brandsen, S. Verberne, K. Lambers, and M. Wansleeben, “Can BERT dig it? named entity recognition for information retrieval in the archaeology domain,” *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 15, no. 3, pp. 1–18, 2022, doi: 10.1145/3497842.

[11] J. Devlin, “Multilingual BERT,” *GitHub*, https://github.com/google-research/bert/blob/master/multilingual.md (accessed Feb. 01, 2023).

[12] B. Wilie *et al.*, “IndoNLU: benchmark and resources for evaluating Indonesian natural language understanding,” in *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, 2020, pp. 843–857.

[13] Y. Liu *et al.*, “RoBERTa: a robustly optimized BERT pretraining approach,” in *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, 2021, pp. 1218–1227.

[14] A. Conneau *et al.*, “Unsupervised cross-lingual representation learning at scale,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8440–8451, doi: 10.18653/v1/2020.acl-main.747.

[15] N. Aletras *et al.*, “Predicting judicial decisions of the European court of human rights: a natural language processing perspective,” *PeerJ Computer Science*, vol. 2, 2016, doi: 10.7717/peerj-cs.93.

[16] M. Medvedeva, M. Vols, and M. Wieling, “Using machine learning to predict decisions of the European court of human rights,” *Artificial Intelligence and Law*, vol. 28, no. 2, pp. 237–266, 2020, doi: 10.1007/s10506-019-09255-y.

[17] K. Kowsrihawat, P. Vateekul, and P. Boonkwan, “Predicting judicial decisions of criminal cases from Thai supreme court using Bi-directional GRU with attention mechanism,” in *2018 5th Asian Conference on Defense Technology (ACDT)*, 2018, pp. 50–55, doi: 10.1109/ACDT.2018.8592948.

[18] M. B. L. Virtucio *et al.*, “Predicting decisions of the Philippine supreme court using natural language processing and machine learning,” in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, 2018, vol. 02, pp. 130–135, doi: 10.1109/COMPSAC.2018.10348.

[19] E. Q. Nuranti, E. Yulianti, and H. S. Husin, “Predicting the category and the length of punishment in Indonesian courts based on previous court decision documents,” *Computers*, vol. 11, no. 6, 2022, doi: 10.3390/computers11060088.

[20] V. Barriere and A. Fouret, “May i check again? – a simple but efficient way to generate and use contextual dictionaries for named entity recognition. Application to French legal texts,” in *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, 2019, pp. 327–332.

[21] E. Leitner, G. Rehm, and J. Moreno-Schneider, “Fine-grained named entity recognition in legal documents,” in *Semantic Systems. The Power of AI and Knowledge Graphs*, 2019, pp. 272–287, doi: 10.1007/978-3-030-33220-4_20.

[22] F. A. Correia *et al.*, “Fine-grained legal entity annotation: a case study on the Brazilian supreme court,” *Information Processing & Management*, vol. 59, no. 1, 2022, doi: 10.1016/j.ipm.2021.102794.
[23] Indonesian Supreme Court, “Direktori Putusan Mahkamah Agung (Directories of Decisions of Supreme Court),” https://putusan3.mahkamahagung.go.id/ (accessed Feb. 01, 2023).
[24] H. Nakayama, “doccano,” GitHub, 2018. https://github.com/doccano/doccano (accessed Feb. 01, 2023).
[25] J. R. Landis and G. G. Koch, “The measurement of observer agreement for categorical data,” *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977, doi: 10.2307/2529310.
[26] K. Sechidis, G. Tsoumakas, and I. Vlahavas, “On the stratification of multi-label data,” in *Machine Learning and Knowledge Discovery in Databases*, 2011, pp. 145–158, doi: 10.1007/978-3-642-23808-6_10.
[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 Conference of the North*, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
[28] G. Lample and A. Conneau, “Cross-lingual language model pretraining,” in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019, pp. 7059–7069.
[29] W. Wongso, C. K. Wah, S. Rahmadani, and S. Limcorn, “Flax-community/Indonesian-roberta-base,” *huggingface*, 2021. https://huggingface.co/flax-community/indonesian-roberta-base (accessed Feb. 01, 2023).
[30] Z. Zhai, D. Q. Nguyen, and K. Verspoor, “Comparing CNN and LSTM character-level embeddings in BiLSTM-CRF models for chemical and disease named entity recognition,” in *Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*, 2018, pp. 38–43, doi: 10.18653/v1/W18-5605.
[31] M. Yin, P. Zhou, T. Xu, and J. Jiang, “Extracting actors and use cases from requirements text with BiLSTM-CRF,” *Journal of Intelligent & Fuzzy Systems*, vol. 44, pp. 4285–4299, 2022, doi: 10.3233/JIFS-221094.
[32] C. Li and K. Ma, “Entity recognition of Chinese medical text based on multi-head self-attention combined with BILSTM-CRF,” *Mathematical Biosciences and Engineering*, vol. 19, no. 3, pp. 2206–2218, 2022, doi: 10.3934/mbe.2022103.
[33] E. F. Tjong Kim Sang and F. De Meulder, “Introduction to the CoNLL-2003 shared task: language-independent named entity recognition,” in *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 2003, pp. 142–147.

BIOGRAPHIES OF AUTHORS

Evi Yulianti is a lecturer and researcher at Faculty of Computer Science, Universitas Indonesia. She received the B.Comp.Sc. degree from the Universitas Indonesia in 2010, the dual M.Comp.Sc. degree from Universitas Indonesia and Royal Melbourne Institute of Technology University in 2013, and the Ph.D. degree from Royal Melbourne Institute of Technology University in 2018. Her research interests include information retrieval and natural language processing. She can be contacted at evi.y@cs.ui.ac.id.

Naradhipa Bhary received his bachelor’s degree in computer science from the Universitas Indonesia in 2023. His research interests are related to natural language processing. He can be contacted at naradhipa.mahardhika@ui.ac.id.

Jafar Abdurrohman received his bachelor’s degree in computer science from the Universitas Indonesia in 2023. His research interests are related to natural language processing. He can be contacted at jafar.abdurrohman@ui.ac.id.

Fariz Wahyuzan Dwitilas received his bachelor's degree in computer science from the Universitas Indonesia in 2023. His research interests are related to natural language processing. He can be contacted at email: fariz.wahyuzan@ui.ac.id

Eka Qadri Nuranti is a lecturer and researcher at Institut Teknologi Bacharuddin Jusuf Habibie. Her research interest includes information retrieval, natural language processing, law AI, and deep learning. She received her bachelor's and master's degree in computer science from the Universitas Indonesia in 2016 and 2020. She can be contacted at email: eka.qadri@ith.ac.id.

Husna Sarirah Husin is a lecturer and researcher at of Innovation and Technology, Taylor's University, Malaysia. She received her bachelor's degree from Conventry University, her master's degree from Universiti Teknologi MARA, and her doctoral degree from Royal Melbourne Institute of Technology. Her research interests include web mining, web analytics, and information systems. She can be contacted at email: husna.husin@taylors.edu.my.