computers

MDPI

Article

Predicting the Category and the Length of Punishment in Indonesian Courts Based on Previous Court Decision Documents

Eka Qadri Nuranti [1,2], Evi Yulianti [1,*] and Husna Sarirah Husin [3]

[1] Faculty of Computer Science, Universitas Indonesia, Depok 16424, Indonesia; eka.qadri91@ui.ac.id or ekaqadri@gmail.com

[2] Institut Teknologi Bacharuddin Jusuf Habibie, Parepare 91125, Indonesia

[3] Malaysian Institute of Information Technology, Universiti Kuala Lumpur, Kuala Lumpur 50250, Malaysia; sarirah@unikl.edu.my

* Correspondence: evi.y@cs.ui.ac.id

Abstract: Among the sources of legal considerations are judges' previous decisions regarding similar cases that are archived in court decision documents. However, due to the increasing number of court decision documents, it is difficult to find relevant information, such as the category and the length of punishment for similar legal cases. This study presents predictions of first-level judicial decisions by utilizing a collection of Indonesian court decision documents. We propose using multi-level learning, namely, CNN+attention, using decision document sections as features to predict the category and the length of punishment in Indonesian courts. Our results demonstrate that the decision document sections that strongly affected the accuracy of the prediction model were prosecution history, facts, legal facts, and legal considerations. The prediction of the punishment category shows that the CNN+attention model achieved better accuracy than other deep learning models, such as CNN, LSTM, BiLSTM, LSTM+attention, and BiLSTM+attention, by up to 28.18%. The superiority of the CNN+attention model is also shown to predict the punishment length, with the best result being achieved using the 'year' time unit.

Keywords: prediction; punishment category; punishment length; court decision document; Indonesian courts; convolutional neural network; attention

check for updates

Citation: Nuranti, E.Q.; Yulianti, E.; Husin, H.S. Predicting the Category and the Length of Punishment in Indonesian Courts Based on Previous Court Decision Documents. *Computers* **2022**, *11*, 88. https://doi.org/10.3390/computers11060088

Academic Editor: Fernando Bobillo

Received: 28 February 2022
Accepted: 19 April 2022
Published: 30 May 2022

1. Introduction

The legal system can be classified according to two types, namely, common and civil law systems [1–3]. The common law system is oriented toward cases (i.e., case law) in which a legal practitioner in a court refers to a previous judges' decisions (i.e., jurisprudence) [3] in resolving a legal problem. By contrast, the civil law system adopts a codification system (i.e., codified law) in which legislation is the primary reference for deciding a particular case.

Indonesia adheres to a civil law system. However, in Indonesia, certain judges also use jurisprudence in addition to the constitution and written legislation to make decisions about a particular case [3,4]. In this case, previous court decisions (i.e., jurisprudence) concerning a similar legal case can be used by judges as a basis for legal decisions. In addition, if there is a legal vacuum (i.e., a condition in which there is no regulation for a particular case), the previous judge's decision should also become a legal instrument to maintain legal certainty [3]. Referring to Butt [5], jurisprudence from the Constitutional Court and the Supreme Court can foster legal consistency and transparency.

According to statistics taken from court decision documents on the Indonesian Supreme Court Decision's website, https://putusan3.mahkamahagung.go.id/ that was accessed on 1 September 2019, the number of Indonesian court decision documents has experienced consistent growth of approximately 100,000 documents each month [6]. Therefore, it is cumbersome to read and interpret each document that is related to a given legal case to consider how heavy the case is compared to the previous cases. This indicates the need for a system that can process the collection of decision documents automatically for the required information to be obtained quickly.

In this work, we utilized a collection of Indonesian court decision documents to conduct a prediction of the category and the length of punishment in similar cases. We argue that this prediction system, together with a legal search engine, may be of benefit to people who work with legal matters. This system could be used by judges as a supporting reference for deciding the appropriate punishment. Furthermore, it could be used by legal actors in the government or scholars to supervise and/or criticize the consistency of decisions made by judges regarding similar cases. In the past few years, several works have studied the use of decision documents written in English [7,8], Filipino [9], and Thai [10] language to predict the decision of a given legal case (guilty/not) using machine learning and deep learning approaches. Our work was different to them in that we did not predict the judicial decision, but we predicted the category and the length of punishment for a given legal case based on previous similar cases archived in the decision documents. To the best of our knowledge, none of the previous work, in any language, studied this problem.

To tackle this problem, we propose using a multi-level deep learning method, that is, the convolutional neural network with attention mechanism (CNN+attention), and the use of sections of the court decision documents as features to predict the category and the length of punishment for a given legal case. This method consists of feature-level learning using the CNN method and document-level learning using the attention mechanism. This method is aimed at improving the accuracy of the predictions by highlighting important information from each feature that affects predictions and then conveying relevant information among the features to the document's level. This method has been shown to have superior performance in previous work on prediction/recognition task [11–13]. However, none of the past works have explored the use of this method for prediction tasks in the legal domain.

The use of sections in the decision documents to learn a prediction model in the legal domain has been studied in some past works [7,8,10]. However, the purpose of their models is for predicting the judicial decision. Therefore, the use of document section features for predicting the category and the length of punishment has not been investigated. In addition, the decision document sections used in our work were also different to them, since we used the collection of Indonesian decision documents, which have different sections than the collections used in these previous works. While some of these studies only used partial sections of the decision documents, we used all sections in the documents and further analyzed which sections were important for our prediction model.

In summary, the contribution of this work is threefold: (1) We explore a novel problem of predicting the category and the length of punishment of a particular legal case based on the previous court decision documents. (2) We propose the use of a multi-level deep-learning method, that is, the convolutional neural network with attention mechanism (CNN+attention), and the use of sections of decision documents as features to build an accurate model for predicting the category and the length of punishment of Indonesian courts. We further analyze which sections in the documents can improve the effectiveness of the prediction model. (3) We performed empirical evaluations on the effectiveness of our system using the CNN+attention method and document sections features on the collection of Indonesian court decision documents against several baselines such as CNN, LSTM, BiLSTM, LSTM+attention, and BiLSTM+attention.

Finally, this work aimed to answer the following research questions:

(1) What information from court decision documents are valuable/important for predicting the category and the length of punishment in Indonesian courts?

(2) How effective is the multi-level learning CNN method with an attention mechanism for predicting the category and the length of punishment in Indonesian courts?

2. Related Work

Research has been conducted to produce useful information from court decision documents [6–14]. Aletras et al. [7] predicted the decision for a legal case (i.e., violation and no violation) by building a binary classification model using English-language court decision documents. Each decision document was mapped into six parts. They used the machine learning algorithm, SVM (support vector machine), and textual features, such as n-grams and topics of each part of the document, to build the classification model. The results revealed that their model achieved an accuracy rate of 79% and that the 'facts' part played an important role in increasing the accuracy of the predictions. Medvedeva et al. [8] used the same data set and classification algorithm as Aletras et al. but only considered the procedure and the facts sections of the documents. The prediction results showed that SVM successfully predicted 75% of all the cases correctly.

Similar to the work described above, Virtucio et al. [9] and Kowsrihawat et al. [10] also predicted the decision of a criminal case (that is, guilty or not guilty) using court decision documents. They, however, used the document collection written in other languages, Filipino and Thai, respectively. Virtucio et al. [9] exploited n-grams and topic features for machine learning algorithms (e.g., SVM and random forest). Their best results were obtained using a random forest classification algorithm with an accuracy of 59%. In contrast to all the work above using machine learning techniques, Kowsrihawat et al. [10] used an end-to-end deep learning model, bidirectional gated recurrent unit (Bi-GRU), with an attention mechanism to make the prediction. They used the facts and law sections of the documents as features for their model. They found that the model could outperform the machine learning baselines: SVM and naive Bayes algorithms. Wu et al. [14] added domain knowledge information combined with machine learning techniques. They found that the domain knowledge that had been created could increase the accuracy of machine learning.

Our work is different from all of the aforementioned work in terms of the problem, task, data set, and methods. **First**, regarding the research problem, while all these previous works studied the prediction of the judicial decision of legal cases (guilty/not guilty), we investigated a different problem: predicting the category and the length of punishment of legal cases. To the best of our knowledge from our extensive literature reviews, we could not find any previous work that studied this problem. **Second**, regarding the task, in contrast to these previous works, which adopted the binary classification task (guilty/not guilty), our task was formulated as (1) a multiclass classification task (mild/moderate/heavy/very heavy) for the punishment category prediction; (2) a regression task for the punishment length prediction. **Third**, regarding the data set, while these previous works used English, Thai, and Filipino data sets, we used an Indonesian data set of court decision documents. This difference impacts on different language characteristics and different sections of decision documents. An example of the latter case is that we identified 11 sections contained in Indonesian decision documents; however, in the English decision documents used by Aletras et al. [7], there were only six sections identified in the documents. **Fourth**, regarding the method, we used different methods from previous work, a multi-level deep learning model, CNN+attention, to build the prediction models.

A few studies of the legal domain have also been conducted using Indonesian court decision documents [6,15,16]. Most of these studies, however, focus on the entity recognition tasks. Solihin and Budi [15] detected important entities in court decision documents for criminal theft cases only using the rule-based approach. Later, Nuranti and Yulianti [6] also extracted some legal entities from decision documents but using machine learning and deep learning methods such as the bidirectional long short-term memory (BiLSTM) and the conditional random field (CRF) approaches. In later work, Violina and Budi [16] attempted to develop an information extraction system for Indonesian law documents using a knowledge engineering approach. In contrast to these works, we predicted the length and the category of punishment for a criminal case using sections of court decision documents as the features for the CNN+attention method.

3. Prediction Model

There were two main tasks conducted in this work: punishment category prediction and punishment length prediction. While the former was formulated as a multiclass classification task, the latter was formulated as a regression task. We used the same method to build the prediction model for the first and second tasks, the convolutional neural network with attention mechanism (CNN+attention). The difference is that in the second task, the dimension value for the CNN+attention model at the document-level learning, which produced the category output, was changed to one dimension to produce a regression output (instead of a category output).

Figure 1 illustrates our prediction model using a multi-level learning method, CNN+attention. The multi-level learning method consists of two basic processes: feature learning and document learning processes. The feature learning process was conducted using the CNN method to learn a model for each feature that can highlight important information in each feature. If there are *n* features, then there will be *n* feature models learned. Next, the document learning process was performed using an attention mechanism to learn a model to combine the result/score given by each feature model. Note that our method is different from a multi-channel deep learning method that involves multiple types of input features with different treatments, such as the method used by Chen [17], which uses a different type of word representation in each channel. Meanwhile, our method is a multi-level deep learning method that uses multiple levels in the learning process (feature-level and document-level learning processes).

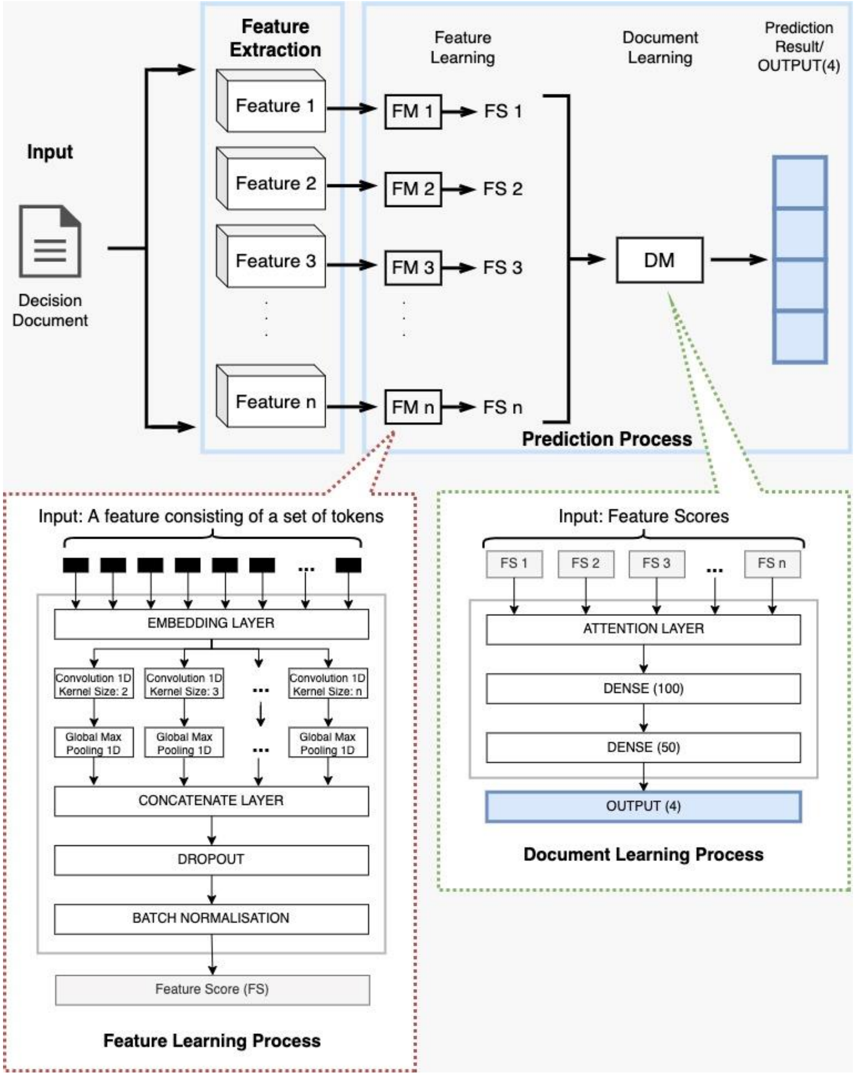


Figure 1. An illustration of the CNN+attention method (FM stands for Feature Model, and DM stands for Document Model).

The court decision documents will pass through the text extraction, token normalization, and document section annotation to perform the feature extraction process. The text extraction was performed to obtain text from court decision documents in PDF format. The token normalization was aimed to normalize the error tokens resulting from the automatic text extraction. Next, document section annotation was conducted to identify and extract text from each section of the decision documents used as features in our prediction model. Since we identified ten sections contained in an Indonesian court decision document (except the verdict section), we had ten features to use in the prediction model. A detailed explanation of the feature extraction process is described in Section 4.

After features were extracted, we then performed a feature learning process using the CNN method [18,19] to learn a model for each feature. This process aimed to capture important information from each feature. CNN has been shown to achieve good performance in the text classification task [20–22] and is able to capture important information contained in each feature [23]. The input for the feature learning process was a set of tokens for each feature that will pass through the embedding layer to obtain a previously trained word vector representation. We used Word2vec [24] and FastText [25] to generate word vector representations as they can capture semantic information. The vector representations are then inputted into the max-pooling [26] layer stage to reduce the overfitting [27], and the output of this layer is called convoluted feature. This stage can bring up information that is considered important during the training period.

Further, each piece of convoluted feature is combined in a concatenate layer [28], which is called a fully connected layer [18]. To conclude the feature learning stage, we used batch normalization [29] to enable the model to retransform the feature model (FM) generated in this feature learning process and ensure that the FM was standardized. The result of the batch normalization was a vector score which we call the feature score (FS). Since there were ten features extracted as described above, this process generated ten FSs.

After the FMs are learned, the next step was to reassemble these feature scores as document information. We called this step the 'document learning process'. When combining all the important information from all the features, we wanted the model to find the relationship for each feature learning result and to rearrange the most relevant information in the overall features used. For this reason, we did not use a concatenate layer in the document learning step; instead, we used the attention mechanism method. The attention mechanism is a concept that was initially found in the encoder–decoder architecture in the translation model [30]. This method is able to determine influential and interrelated information between features in parallel [31,32] and is also efficient in terms of memory usage. The latter reason was more important, considering that we used 10 features in our prediction method. The result of the attention layer passed through dense layers of sizes 100 and 50, because we needed the nodes to be fully connected so that it was easier to make predictions [33]. The final step used a dense layer with four dimensions (array sizes) as the output of the prediction of the punishment category (i.e., mild, moderate, heavy, and very heavy). In the task of the prediction of punishment length, the dimension value was changed to one dimension in order to produce a regression output (instead of a category output).

4. Research Methodology

This section is divided into five subsections. The first subsection explains the data collection process and the distribution of the data set. The second and third subsections describe the data pre-processing, together with the section annotation process. The last section reviews all the experiments that were conducted in this work.

4.1. Data Collection

Each decision on the Indonesian Supreme Court's website is accompanied by metadata that is also provided on the website. Some of the information contained in the metadata are the URL, the document number, the province, the district court institution, the case level, the case classification, the verdict (the length of punishment), the document status, and the pdf document file. Not all decision documents were used in this research. The following criteria were required for our data set:

1. First-degree criminal decisions;
2. Decisions that had permanent legal force (*inkracht*) [4] that were obtained from district courts in West Java, Central Java, East Java, Jakarta, or Yogyakarta;
3. Decision documents that had pdf files.

We crawled the metadata of 82,827 decision documents. We could not use all of them because the distribution of documents was uneven, in which there are approximately 74% of the decisions with the length of punishment between 1–1000 days. Therefore, of these decision documents, we randomly selected 5000 documents with the punishment length between 1–500 days, and 5000 documents with the punishment length between 501–1000 in order to balance the data set. The rest of the documents with the punishment length greater than 1000 days were all used. After this filtering process, the total number of documents used in this study was 22,630.

Figure 2 shows the distribution of the length of punishment (converted into days) in our data set. The length of punishment ranged from zero days (free/exempt from punishment) to 8000 days. This figure shows that the data were not balanced, particularly in data distributions when the prison sentence was longer than 2000 days. All the decision documents were then grouped based on the distribution of the quartile length of the criminal punishment for each decision document. We used four categories of punishments in this work: mild, moderate, heavy, and very heavy. A document with a length of punishment that was less than the first quartile (Q1) was classified as 'mild', between the first quartile (Q1) and the second quartile (Q2) as 'moderate', and between the second quartile (Q2) and the third quartile (Q3) as 'heavy', while punishments that exceeded the third quartile (Q3) were categorized as 'very heavy'. In this case, the values for Q1, Q2, and Q3 were 480, 1080, and 1800, respectively.

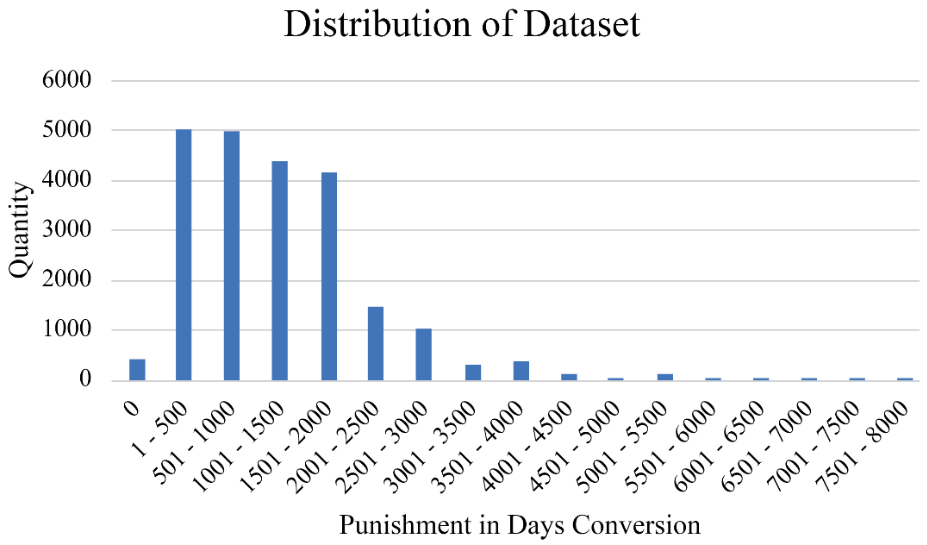


Figure 2. The distribution of decision documents in our data set was based on the length of punishment (in 'day' units).

The number of court decision documents for each category can be seen in Table 1. In general, the number of documents for each category was quite balanced. The category with the highest number of documents was 'moderate', while the category with the smallest number of documents was 'heavy'. We selected 10% of them as test data, and the rest was used for training.

Table 1. The distribution of decision documents in our data set for each punishment category.

Punishment Category	Punishment Length (Days)	Number of Documents
Mild	0–479	5561
Moderate	480–1079	5991
Heavy	1080–1799	4112
Very Heavy	1800–8000	5811

4.2. Feature Extraction

This section describes the process to extract features from the decision documents. It involved three steps: text extraction, token normalization, and section annotation.

4.2.1. Text Extraction

Indonesian court decision documents are accessible in pdf format on the Indonesian Supreme Court Decision's website. Watermarks, headers, footers, and page formats were all unnecessary formatting elements on each document page. All the undesired elements of the pdf document were deleted, leaving only the verdict's plain text. The Python library, PyMuPdf, https://pymupdf.readthedocs.io/ that was accessed on 1 October 2019, was used to convert pdf documents into text format.

4.2.2. Token Normalization

In this stage, the tokens/terms in each court decision document were normalized into normal form. This stage is important because many error/invalid tokens were caused by converting pdf files into text fields [6,15]. This step was expected to increase the number of valid tokens to improve the accuracy of our learning models in the training process. To obtain the tokens to be normalized, we found tokens that appeared less than ten times in a document. We chose ten as the minimum number of occurrences for each token and argue that this number was sufficient to identify likely invalid tokens (for example, typos or multiple words that merged into a single word due to the fact of missing spaces) since they rarely appeared in the documents. Tokens that could not be normalized were considered as unknown tokens. We kept the unknown token in this process to keep the flow of text in the document unchanged.

The initial number of tokens from all documents in the data set was 970,214 tokens. Using the token filtering criteria discussed above, 222,180 tokens fulfilled our requirements because they occurred a minimum of 10 times in a document. Therefore, they were used as our dictionary. The remaining 748,034 tokens that did not satisfy our requirements were then subjected to the normalization process. We did not remove these words immediately because we could still obtain their valid forms via normalization. The following steps were used for the token normalization:

- We identified the tokens that were typos or had the same meaning by examining the smallest edit distance compared to tokens in the dictionary, which contained 222,180 tokens. If the minimum edit distance obtained was less than three, it was considered as a typo. Approximately 96% of the tokens could be identified in this step; for example, 'fundamentum' (*fundamental*) was normalized as 'fundamental' (*fundamental*);
- If the minimum edit distance was more significant than three, we checked for invalid words that occurred because space was missing. Beginning from the left-hand side, we traced the word letter by letter and then checked whether the combination of letters existed in the dictionary. If the combination existed, we inserted a space and continued to trace the remaining letters in the word. We proceeded from the right-hand side if we could not search from the left-hand side. Approximately 26,676 tokens were identified and separated by spaces in this step; for example, 'tidaknyapadasuatuwaktu' was normalized as 'tidaknya pada suatu waktu' (*not at a time*);

- If the above step was unsuccessful, the next step was to identify the token as an unknown token by mapping it as a token '_unk_'. For example, 'jpoooooooporoorjo' was '_unk_'.

4.2.3. Document Section Annotation

Court decision documents contain some common information, such as a document opener, the defendant's identity, the case history and the like [1]. However, this information is not presented clearly in sections; instead, it forms part of the complete text in a decision document. Therefore, in order to identify sections for each information category in the document, and to further examine which document sections were useful for our prediction task, we needed to perform document section annotation.

In this process, the sections of court decision documents were used as features to predict the category and the length of punishment of a criminal case. We aimed to determine which parts of the decision documents had the greatest effect on the prediction results. It was intriguing to note which sections were useful for predicting the punishment category and length in a criminal case and which were not.

Table 2 presents all the document sections that we identified exist in the court decision documents. The third column in the table highlights the chunks of text contained in each form; it follows the template provided in Keputusan Mahkamah Agung Nomor 44 Tahun 2014 (*the Decision of the Chief Justice of the Supreme Court of the Republic of Indonesia Number 44 of 2014 concerning the Enforcement of Decision Templates and Numbering Standards for General Court Cases*), that was accessed from https://badilum.mahkamahagung.go.id/berita/pengumuman-surat-dinas/2022-sk-kma-nomor-44-tahun-2014-tentang-pemberlakuan-template-putusan-dan-standar-penomoran-perkara-peradilan-umum.html on 8 April 2022. This document is referred to as 'decision document number 44/KMA/SK/III/2014' hereafter.

The document opener includes information about the document's title, starting with the word 'PUTUSAN' ('*DECISION*'), the verdict number, the sentence 'DEMI KEADILAN BERDASARKAN KETUHANAN YANG MAHA ESA' ('*FOR JUSTICE BASED ON ONE ALMIGHTY GOD*'), and a description of the case. The defendant's identity discloses general information about the defendant, such as name, place of birth, age, date of birth, gender, nationality, residence, religion, occupation, and last education. The case history was divided into three parts, namely, detention history, indictment history, and prosecution history.

Information related to the processes of a court case is described in the facts, legal facts, and legal considerations sections. The facts section consists of witnesses' statements, experts' statements, defendants' statements, letters, instructions, tools, and evidence. Legal facts are essential to the prosecutor's point and contain the relationship among the facts; legal considerations include the judges' deliberations in determining a case based on the existing legal facts.

There are two parts at the end of the verdict, namely, the verdict and the verdict's closing. The verdict contains the judge's decision regarding the case and includes the length of punishment if the defendant is proven to have committed a criminal act. The verdict's closing consists of the day, the date, the year, the judges who decided on the case, the court clerk, the signatures of the panel of judges, and the cost of the case. Since the verdict sections contain information about the length of punishment, we could not use verdict section as a feature. Note that the length of punishment is the key element to be predicted by our model in this work. Therefore, our prediction model used 10 features in total (all sections listed in Table 2 except the verdict section).

It is important to note that none of the Indonesian court decision documents had an annotation structure. Moreover, there was no precise string pattern for each section in the documents; there was no clear division between one section and another. Due to the fact of this condition, we asked people who were familiar with the legal domain to annotate each section in a decision document.

Table 2. The sections in the court decision documents and the strings that often identified the sections.

No.	Document Sections	Strings That Often Identified the Sections
1.	Kepala putusan (*document opener*)	'PUTUSAN' ('*DECISION*') Always in the first line
2.	Identitas terdakwa (*defendant's identity*)	'Nama . . . ', 'Terdakwa I: Nama . . . ' ('*Name . . . ', "Defendant I: Name . . . "*)
3.	Riwayat perkara (*case history*)	'Para terdakwa didampingi oleh penasihat . . . ', 'Pengadilan Negeri tersebut' ('*The defendants were accompanied by an adviser . . .* ')
4.	Riwayat penahanan (*detention history*)	'Terdakwa ditahan dengan penahanan Rumah/Rutan/Kota/Negara oleh . . . ', 'Terdakwa dalam perkara ini tidak ditahan' ('*The defendant was detained at Home/Detention Center/City/State detention by . . . ', 'The defendant in this case was not detained*')
5.	Riwayat tuntutan (*prosecution history*)	'Telah/Setelah mendengar tuntutan . . . ' ('*After hearing the demands . . .* ')
6.	Riwayat dakwaan (*indictment history*)	'Menimbang, bahwa terdakwa . . . dakwaan Jaksa . . . ' ('*Considering, that the defendant . . . the prosecutor's indictment . . .* ')
7.	Fakta (*facts*)	'Menimbang, bahwa dipersidangan telah menga-jukan/mendengar/membaca/memeriksa . . . ' ('*Considering, that at the court submitted/heard/read/examined . . .* ')
8.	Fakta hukum (*legal facts*)	"Menimbang, . . . fakta-fakta/fakta hukum . . . " ("*Considering, . . . facts/legal facts . . .* ")
9.	Pertimbangan hukum (*legal considerations*)	'Menimbang, . . . majelis hakim . . . berdasarkan fakta hukum/fakta-fakta . . . ' ('*Considering, . . . the panel of judges . . . based on facts/legal facts . . .* ')
10.	Amar putusan (*verdict*)	'MENGADILI' ('*JUDGE*')
11.	Penutup (*closing*)	'Demikianlah . . . ' ('*Declares . . .* ')

The manual annotation involved two annotators to annotate the sections in 1000 documents. Our annotators were two students who were law majors; thus, they were familiar with the content of court decision documents. Each annotator performed annotations on 550 documents, with 100 of them overlapping between the two annotators. These overlapping documents were necessary in order to compute the agreement score between the two annotators.

We built a web-based annotation tool to make the annotation job easier; the annotators were first trained to use the system before they began their annotation work. We presented the annotators with the guidelines for annotating a first-degree case decision; these guidelines were obtained from decision document number 44/KMA/SK/III/2014 concerning the Enforcement of Decision Templates and Standard Numbering Cases for General Courts. For each document to be annotated by our annotators, we divided the document into sentences and displayed each sentence on one line. Section annotation was performed by specifying the starting and ending line numbers of the sentences that belong to each section. Our annotation tool will automatically segment the text in each section after obtaining from annotators the line numbers of starting and ending sentences for each section.

The cost of performing a manual annotation is high, which caused us to be unable to perform manual annotations for all the documents in our data set. Recall that our data set consisted of 22,630 decision documents. As we only annotated 1000 documents, the remaining 21,630 documents still did not have annotations in the document section. Accordingly, we performed a rule-based approach to annotate the remaining documents using Python's regular expression (RegEx). This rule-based annotation was based on the template guide in decision document number 44/KMA/SK/III/2014 using the pattern terms specified in Table 2.

To test the effectiveness of our rule-based approach to automatically annotate the sections from decision documents, we calculated the kappa agreement between the text extracted from each section using the automatic rule-based approach and the ones extracted manually by annotators for the 100 overlapping documents. This computation is aimed to examine the feasibility of using automatic rule-based approach for section annotations, by examining how well the automatic annotation process performed compared to the annotation process conducted by humans. Table 3 shows the agreement results for the two annotators, and the agreement between the rule-based annotation and the human annotation that was measured using Cohen's kappa.

Table 3. The results of the annotators' agreement in the document section annotation.

Annotator	Cohen's Kappa
1 and 2	0.93
1 and *Rule-based*	0.71
2 and *Rule-based*	0.70

We can see from the table that the agreement between the two annotators was very high, resulting in a kappa score of 0.93. According to Krippendorff [34,35], the agreement level of this score reflects 'almost perfect' agreement. The agreement between the rule-based annotation and the two annotators was lower than the human annotators, but it was still satisfactory, as they belong to the 'substantial' agreement level [34,35]. Because the agreement between the rule-based annotation and the human annotation was quite high, we argue that applying the rules to the remainder of the documents was acceptable. Based on this result, the annotation process described above were then used to extract the sections of decision documents to be used as features for our multi-level deep-learning model, CNN+attention. The data set resulted by this annotation process, called indo-law data set, has been made available for research purposes at https://github.com/ir-nlp-csui/indo-law.

4.3. Experiment

This study aimed to predict the category and the length of punishment in a criminal case. Our experiment used all decision documents in our data set (82,827 documents). Table 4 describes the summary of our experiment scenario conducted in this work.

In the **first experiment**, we selected the best feature map to maximize the prediction results. Two different feature maps were tested in our experiment, namely, Word2vec [24] and FastText [25]. Both word representations are widely used techniques [36–40]. Word2vec is an established technique for learning word embedding that is an extension of continuous bag-of-words (CBOW) models and skip-gram models [41]. Both models have shown excellent performance in the NLP domain to date. Hence, Word2vec can map token-to-vector representations considering the context around the token [24]. Moreover, FastText is an enhancement of the training model on Word2vec that was developed by Facebook [42]. The concept of FastText compared to Word2vec entails a more profound recognition of a token by forming n-gram characters [25]. We used Python's Gensim library for both the Word2vec and the FastText models to generate word representation. We compared the accuracy of the models using these two-word representations. Both models were trained using 100 vector dimensions, the default settings derived from the Gensim library's parameters. The word representation with the best accuracy results was used for the rest of the experiment.

Table 4. List of Our Experiment Scenario.

No.	Experiment Scenario	Method	Evaluation Metric
1.	The comparison of the best feature representation	Fast TextWord2vec	accuracy
2.	The comparison of using and not using document section features	CNN+attention (full text) **CNN+attention (document section features)**	accuracy
3.	The investigation of important features	Ablation analysis	accuracy
4.	The prediction of punishment category	LSTM BiLSTM CNN LSTM+attention BiLSTM+attention **CNN+attention**	precision, recall, *F-1* score, and accuracy
5.	The prediction of punishment length	LSTM+attention BiLSTM+attention **CNN+attention**	R2 score

The method printed in boldface is our method.

After the best word representation was obtained, we continued with the **second experiment** to examine the merit of using the extracted features from the document sections. To achieve this, we compared the results obtained when using our document section features and when not using the features (simply using all the document's content). The results of this experiment serve as the motivation for using decision document sections as features for our prediction task.

In the **third experiment**, an ablation study was then conducted to determine which features in the judicial decision documents were helpful in the prediction model. This experiment was conducted by removing any single feature from the model and observing the increase/decrease in the accuracy resulting from this removal. We then used the best combination of features for the rest of the experiment to improve the effectiveness and efficiency of our prediction model.

The **fourth and fifth experiments** were our main experiments in this work to predict the category and the length of punishments using the multilevel deep learning CNN+attention method. The training process was conducted using the CNN+attention model and the 10 document section features explained above (see Sections 3 and 4.2, respectively). In the fourth experiment to predict the punishment category, the dimension value for the model in the document-level learning was set to four, since it will produce the category output and there were four categories specified (mild, moderate, heavy, and very heavy). In the fifth experiment, the CNN+attention model was also used for punishment length prediction, but we converted the output of a category prediction into a regression prediction. This is achieved by changing the dimension value for the model in the document-level learning into one. We then compared the prediction result of punishment length on different time units: day, month, and year. Here, one month was converted into 30 days and one year was converted into 360 days.

The effectiveness of our prediction models was measured by comparing the results of our model with some baselines. In the fourth experiment, there were two types of baseline models that were used for comparison:

- One-level deep learning methods, which combine all the features used in a feature map. These baseline methods included CNN, LSTM, and BiLSTM;
- Multi-level deep learning methods, which consist of feature learning and document learning processes. These baseline methods included the LSTM+attention and BiLSTM+attention.

In the fifth experiment, we only took the multi-level learning methods as the baselines, since they were shown to be superior to the one-level learning methods in the fourth experiment (see Section 5.4).

The evaluation metrics that were used in the fourth experiment were precision, recall, F1 score, and accuracy [35,43]. In the fifth experiment, the evaluation metric used the coefficient of determination (R2 score) [44]. This metric has commonly been used in previous research on the regression task [45]. The coefficient of determination indicates how well the independent variable's contribution to the regression model can explain the variation of the dependent variable [46]. The R2 score value is between 0 and 1. Larger values of R2 usually indicate that the regression model fits the data (prediction and actual value) [47]. In contrast, if the R2 score is low, then the relationship between the actual data and predicted data is low, meaning that the predicted data differ much from the actual data.

5. Results and Discussion

We discuss the experimental results in five subsections. We first present the results for finding the best feature representation in Section 5.1; we compare the use and nonuse of the document section features in Section 5.2 and the identification of important features for our prediction tasks in Section 5.3. In Sections 5.4 and 5.5, we then describe the results for the predictions of the punishment category and the punishment length, respectively.

5.1. The Results for the Comparison of the Best Feature Representation

Table 5 presents the accuracy of our multi-level CNN+attention model using ten document section features to predict the category of punishment. With reference to the table, we can see that using Word2vec for feature representation was more accurate than it was using FastText. In this case, Word2vec outperformed FastText by 4.19%.

Table 5. The comparison of prediction results when using FastText and Word2vec representations.

Model	Accuracy
FastText	72.94%
Word2vec	**77.13%**

Note: The highest scores are printed in boldface.

Word2vec had better accuracy than did FastText because this work did not need to analyze words in the character representations. It should be noted that FastText is an improvement on the training model in Word2vec, in which the word vector is formed from n-gram characters [25]. Since the prediction of the punishment category works at the document level (i.e., document classification task), the more suitable and common the features are of the text that composed the documents, which is represented as a set of tokens (words) [7–11,14]. More specifically, the granularity of a feature's element is at the word level, instead of the character level. A character-level feature is usually used when the task subject is at the word/phrase level; thus, the features are the component of that word/phrase, which is characters. For example, in the word-level language identification task [48], some character-level features were used to identify the language of a certain word.

These findings merely demonstrate that Word2vec produced the best outcomes when it was used as the feature representation in our prediction model. Therefore, we decided to use Word2vec for the feature representation in the rest of our experiments, the results of which are described in the following subsections.

5.2. The Results for the Comparison of Using and Not Using Document Section Features

This experiment was conducted to investigate the accuracy of the CNN+attention model using ten document sections as features compared with the one that did not use them as features. More specifically, in the latter case, we used all content in a document as input for our model.

The results are presented in Table 6; in this table, it can be observed that using ten document sections as features in the model was 54% significantly more accurate than was directly using all the content in the document. This is because there is a maximum token parameter setting in a deep learning method, which limits the maximum number of tokens that can be processed by the model. Thus, when a large-sized document was inputted into the model, it was cut into a smaller document containing only the beginning *n* tokens in the document (*n* denotes the value of the maximum token parameter); therefore, only those tokens that could be processed by the model in the learning process. In contrast, when we used document section features, then all of the text in the document contained in each section could be processed by the model, and the score from each feature will then be combined into the document's score. Consequently, the resulting model was significantly more accurate.

Table 6. The use of document section features vs. not using document section features.

Input	Accuracy
Not using the document section features (that is, using all the content in the document)	50.12%
Using the document section features (10 features)	**77.13%**

Note: The highest scores are printed in boldface.

This result is consistent with previous research that also achieved good performance when using sections of documents as features to build the prediction model in the legal domain [7,8,10,49]. A few differences between these works and our work were in terms of the sections contained in the document and/or the way to divide documents into sections. In the latter case, for example, the prior work in [49] divided documents into sections by simply splitting the whole document uniformly into *n*-chunks.

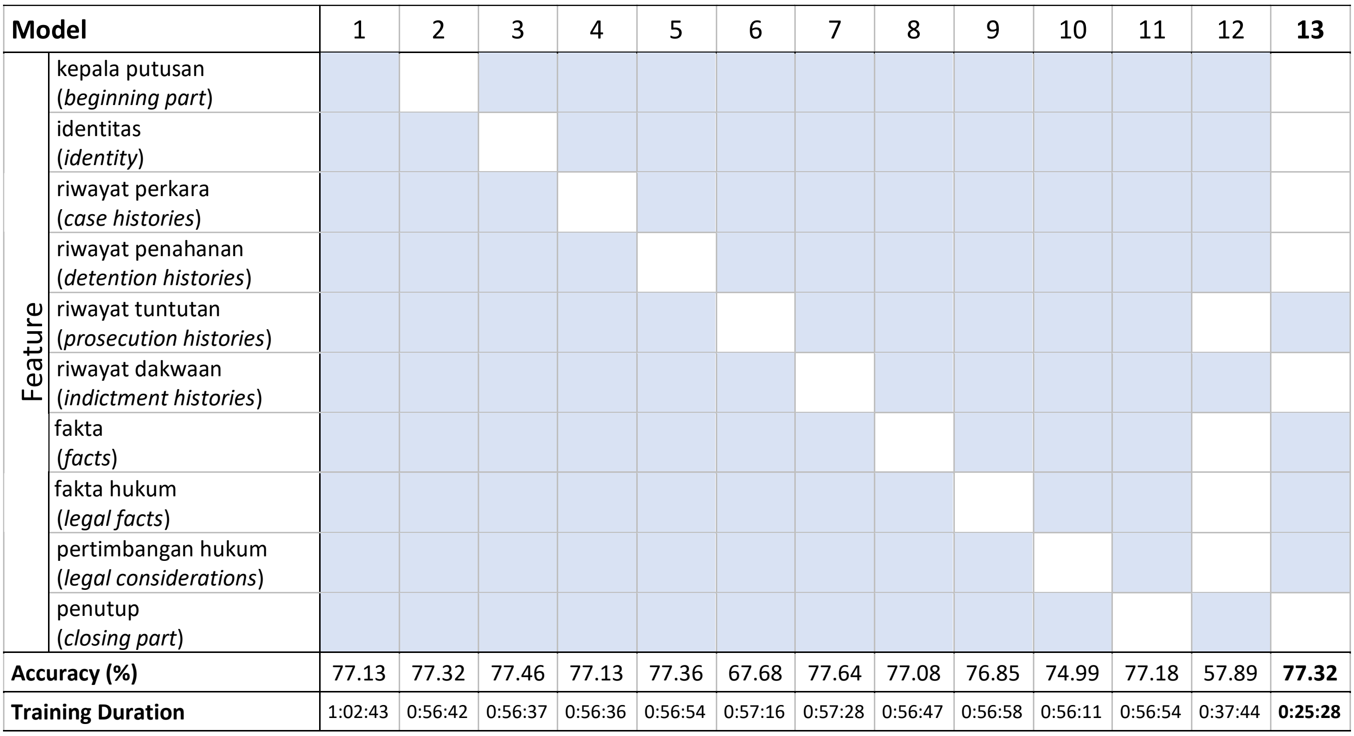
5.3. Ablation Analysis Results

We aimed to examine which features were useful for predicting the category of punishment. Removing features that were useful for prediction would result in decreasing the accuracy of the model. Conversely, removing features that were not useful would result in increasing the accuracy. The results displayed in Figure 3 used the best word representation from the previous experimental results, Word2vec. The features used in the models are marked with a blue cell, while the features that were removed from the models are marked with a white cell.

Model 1, which used all 10 features, obtained an accuracy of 77.13%. From Model 2 to Model 11, a particular feature was removed. It appears from Figure 3 that a decrease in accuracy was observed when the prosecution history, facts, legal facts, and legal consideration section features were removed from the model. Therefore, these four features were considered as useful features for predicting the punishment category. The decrease was up to 9.45% compared to the accuracy of Model 1, which used all 10 features. The most important feature, which resulted in the highest decrease in accuracy, was the claim history.

To examine the degree to which the results decreased when removing all the useful features (i.e., prosecution history, facts, legal facts, and legal consideration), Model 12 was learned. We found a 19.24% decrease compared to the baseline accuracy. This confirmed that these useful features were essential for enhancing the accuracy of the prediction model.

Some features were shown not to be useful for the prediction model, namely, the beginning parts, the identities, case histories, detention histories, indictment histories, and the closing parts. The accuracy of the baseline increased when these features were not used. Therefore, we experimented with learning a model that did not use any of these features (Model 13). There was an increase in accuracy of 0.19% compared to Model 1. Note that, although the score difference against Model 1 was relatively small, Model 13 was more efficient than Model 1 because it used less than half of the features used in Model 1.



Model		1	2	3	4	5	6	7	8	9	10	11	12	**13**
Feature	kepala putusan (*beginning part*)	🟦		🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	
	identitas (*identity*)	🟦	🟦		🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	
	riwayat perkara (*case histories*)	🟦	🟦	🟦		🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	
	riwayat penahanan (*detention histories*)	🟦	🟦	🟦	🟦		🟦	🟦	🟦	🟦	🟦	🟦	🟦	
	riwayat tuntutan (*prosecution histories*)	🟦	🟦	🟦	🟦	🟦		🟦	🟦	🟦	🟦	🟦		🟦
	riwayat dakwaan (*indictment histories*)	🟦	🟦	🟦	🟦	🟦	🟦		🟦	🟦	🟦	🟦	🟦	
	fakta (*facts*)	🟦	🟦	🟦	🟦	🟦	🟦	🟦		🟦	🟦	🟦		🟦
	fakta hukum (*legal facts*)	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦		🟦	🟦		🟦
	pertimbangan hukum (*legal considerations*)	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦		🟦		🟦
	penutup (*closing part*)	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦		🟦	
Accuracy (%)		77.13	77.32	77.46	77.13	77.36	67.68	77.64	77.08	76.85	74.99	77.18	57.89	**77.32**
Training Duration		1:02:43	0:56:42	0:56:37	0:56:36	0:56:54	0:57:16	0:57:28	0:56:47	0:56:58	0:56:11	0:56:54	0:37:44	**0:25:28**

Figure 3. The results for the ablation analysis of the document section features. The blue color denotes that a corresponding feature specified in the table's row is used by a model specified in the table's column. On the contrary, the white color denotes that a corresponding feature specified in the table's row is not used by a model specified in the table's column. The model that achieves the highest accuracy is printed in boldface.

The efficiency of Model 13 can be seen from the duration of the training process. Model 1, which used ten features during training, had a higher number of connected networks than Model 13, which only used four features. The training time for Model 1 was twice that for Model 13. Therefore, in general, Model 13 was preferred over Model 1. For this reason, this model was used in the rest of our experiments.

5.4. The Results for the Prediction of Punishment Category

Table 7 shows the results for predicting the punishment category for each model using a combination of the best features produced by the ablation experiment described above. Here, we used Model 13, which incorporated four useful features, namely, prosecution histories, facts, legal facts, and legal considerations. The table shows that the results for the one-level learning methods were lower than were those for the multi-level learning methods. CNN was shown to be the most effective of all the one-level learning methods. Thus, CNN outperformed the accuracy of the LSTM and BiLSTM methods by 52.97% and 44.47%, respectively. The effectiveness of BiLSTM was slightly better than that of LSTM, because BiLSTM can consider the context at the feature level from two directions, namely, backwards and forwards, in respect to the context [50].

The results for the multi-level learning methods were consistent with those for the one-level learning methods. In this case, the multi-level version of CNN achieved the best results compared to the multi-level versions of LSTM and BiLSTM. The multi-level CNN+attention model outperformed the accuracy of the LSTM+attention model and the BiLSTM+attention model by 19.25% and 18.04%, respectively.

Table 7. The comparison of the effectiveness of CNN+attention method and other deep learning methods in predicting the punishment category.

Model	Precision	Recall	F1Score	Accuracy
LSTM	44.81%	44.23%	41.85%	49.14%
BiLSTM	50.53%	47.95%	43.77%	52.03%
CNN	74.58%	75.10%	74.55%	75.17%
LSTM+attention	65.01%	64.87%	64.80%	65.44%
BiLSTM+attention	65.30%	65.73%	65.43%	65.81%
CNN+attention	**77.08%**	**77.36%**	**76.81%**	**77.32%**

Note: The highest scores are printed in boldface.

The explanation for this result is that referring to the model architecture for LSTM and BiLSTM, each cell state is connected for all the information to be recorded. Therefore, those methods are good at capturing sequence information [51,52]. However, the sequence information is not so crucial in the Indonesian decision documents because these documents have a particular template, which is stated in decision number 44/KMA/SK/III/2014. Many of the phrases are repeated in several lines such as: 'Setelah mendengar . . . ' ('*After hearing* . . . '), 'Menimbang, bahwa terdakwa . . . ' ('*Considering that the defendant* . . . ') or 'Menimbang, bahwa oleh karena . . . ' ('*Considering that because of* . . . '), etc., (see Section 4.2). This implies that the most important information is a piece of data that is filled in after these phrases. Consequently, we only needed to record information that was important to each feature. Since the CNN works based on this concept, which is using a kernel concept to capture information that is important on the feature map and create a convoluted feature, thus, the CNN can capture information that is deemed to be important for each of the features. This explains the reasons why CNN+attention can outperform the LSTM+attention and BiLSTM+attention methods.

Overall, our method using multi-level CNN+attention attained the highest effectiveness across all the metrics. This indicates the effectiveness of our approach to using CNN with multi-level learning. To understand the evaluation results obtained in each category of the multi-level CNN+attention method, we broke down the scores presented in the table above for each category, displayed in Table 8. Based on the F1 score, the 'very heavy' category prediction was the most accurate, followed by the 'mild', 'moderate', and 'heavy' categories.

Table 8. The results of the CNN+attention method in each punishment category.

Class	Precision	Recall	F1score
Mild	79%	80%	79%
Moderate	**80%**	72%	75%
Heavy	76%	64%	69%
Very Heavy	75%	**92%**	**82%**

Note: The highest scores are printed in boldface.

An error analysis was then performed on the prediction results of the multi-level CNN+attention method for each category in order to determine whether a certain category was often misclassified as another category. Figure 4 illustrates the confusion matrix based on the prediction results of the multi-level CNN+attention model. The prediction error caused the 'heavy' category to have the smallest F1 score because it contained numerous prediction errors; for example, 142 documents in the 'heavy' category were misclassified as 'very heavy'.

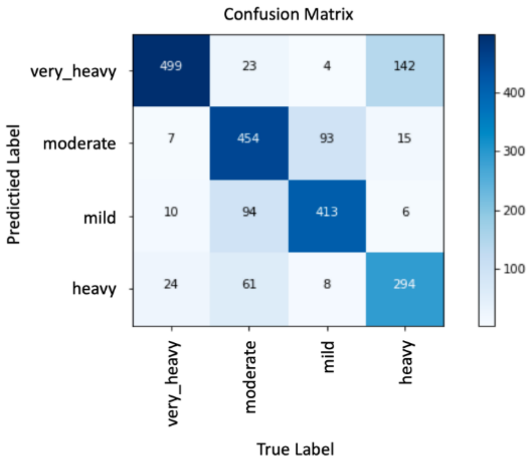


Figure 4. The confusion matrix for the prediction results generated using the CNN+attention model.

5.5. The Results for the Prediction of Length of Punishment

We experimented with three different time units, namely days, months, and years, to predict the punishment length. Table 9 shows the R2 score for the prediction of sentence length based on the day, month, and year time representation using the CNN+attention model. We can see that the best R2 score of 67.16% was obtained by using the year time unit, while the lowest score was obtained by using the day unit of time. This is understandable because the "day" time unit had the greatest range compared to the month and year time unit. Therefore, the likelihood of accurate prediction was the least when using this time representation.

Table 9. The results for the CNN+attention method in predicting the length of the punishment.

Time Unit	Value Distribution	R2 Score
Days	0–8000	44.37%
Months	0–270	60.79%
Years	**0–23**	**67.16%**

Note: The highest scores are printed in boldface.

To examine the effectiveness of CNN+attention in the prediction of the length of the sentence, we also evaluated the results against the baselines. We chose the two strongest baselines from the results for the category prediction mentioned above, namely LSTM+attention and BiLSTM+attention. Table 10 presents the R2 scores for all the models using the year time unit.

Table 10. The comparison of the effectiveness of CNN+attention method and other deep learning methods in predicting the length of punishment.

Model	R2 Score
LSTM+attention	38.65%
BiLSTM+attention	41.54%
CNN+attention	**67.16%**

Note: The highest scores are printed in boldface.

The R2 scores for LSTM+attention and BiLSTM+attention were lower than the score for CNN+attention. Based on these results, we can see that the R2 score results were consistent with the results for the model's accuracy in the punishment category prediction described in Table 8. This may have been because, as mentioned earlier in this section, the prediction for the length of punishment was obtained by changing the output layer of the CNN+attention model that was used to predict the punishment category (from four dimensions to one dimension) to produce the regression output. Therefore, the superiority of the CNN+attention method in the punishment category prediction is also shown in the punishment length prediction.

Figure 5 shows the scatter plots that depict the relationship between two paired data, namely the actual data (on the *x*-axis) and predicted results (on the *y*-axis). The plot will be in the form of a perfectly linear pattern if the lengths of the predicted and actual punishments matched (have a high correlation). If a linear pattern is not displayed on the plot, then the data were not correlated. The plots for the regression results using day, month, and year units are illustrated in plots a, b, and c, respectively.

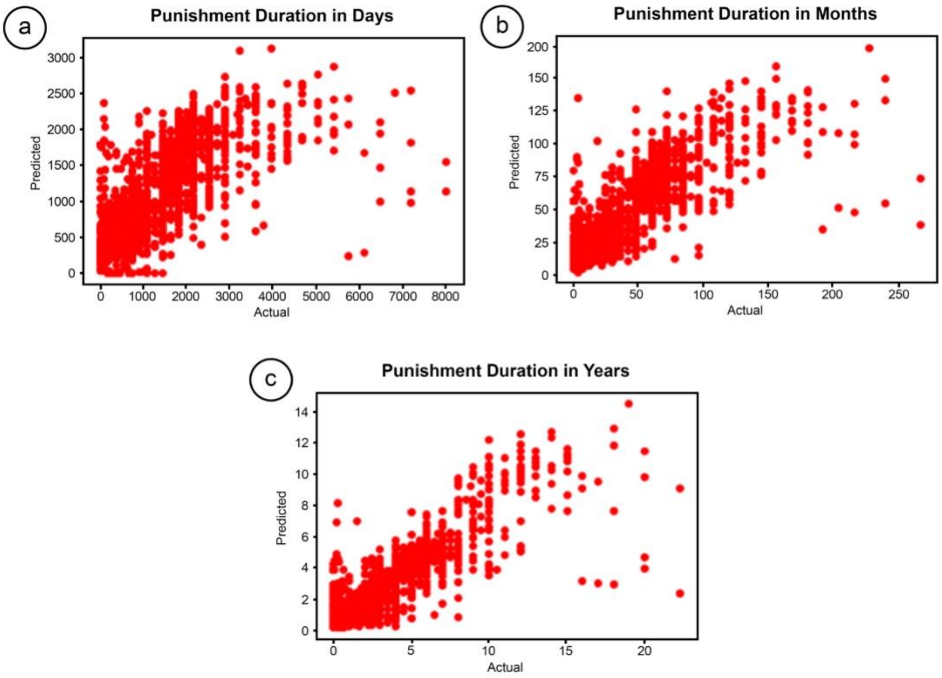


Figure 5. The prediction results for the length of punishment using day, month, and year time units. (**a**) Scatterplot of predicted punishment length using day time unit, (**b**) Scatterplot of predicted punishment length using month time unit, and (**c**) predicted punishment length using year time unit.

The three scatter plots follow the linear data somewhat; thus, they can be moderately linear. The data points in scatter plot c were denser and formed a more significant linear distribution than the data points in scatter plots a and b. This indicates that using a year time unit was more accurate than using day and month time units in predicting the punishment length. This result is supported by the R2 scores displayed in Table 9, which show that the model using the year time unit achieved higher scores than the ones using day and month time units.

6. Conclusions and Future Work

This study examined predictions for the category and length of criminal punishment in Indonesian courts using previous court decision documents. We proposed using a multi-level deep learning method, that is, the convolutional neural network (CNN) approach with an attention mechanism (CNN+attention), and the use of decision document sections as features to build the prediction models. Two main tasks were studied in this work: prediction of the punishment category, which was formulated as a multiclass classification task, and prediction of the punishment length, which was formulated as a regression task.

Our results revealed that using the document section features could significantly improve the model's performance that did not use the features by 54%. The document section features that were important in our prediction tasks were prosecution histories, facts, legal facts, and legal considerations. The proposed multi-level CNN+attention model outperformed all the single-level deep learning baselines (i.e., CNN, LSTM, and BiLSTM) and multi-level deep-learning baselines (i.e., LSTM+attention and BiLSTM+attention) by 2.15% to 28.18% when predicting the category of punishment. This result is consistent with the model's accuracy in predicting the length of punishment. The CNN+attention also significantly outperformed other multi-level deep learning baselines. The best results were achieved when using the year-time representation.

We aim to add additional features that may affect the prediction results in the future. Several features of the law that have not been used in this work include law articles used in resolving cases, the names of the judges and prosecutors involved, the institution's location, and specific categories of crime cases (such as traffic violations, human rights violations, and the like). These features may also influence the judges' decisions.

Author Contributions: Conceptualization, E.Q.N. and E.Y.; Data curation, E.Q.N.; Formal analysis, E.Q.N. and E.Y.; Funding acquisition, E.Y. and H.S.H.; Investigation, E.Q.N. and E.Y.; Methodology, E.Q.N. and E.Y.; Project administration, E.Y. and H.S.H.; Resources, E.Q.N.; Software, E.Q.N.; Supervision, E.Y. and H.S.H.; Validation, E.Q.N. and E.Y.; Visualization, E.Q.N.; Writing - original draft, E.Q.N. and E.Y.; Writing—review & editing, E.Q.N., E.Y. and H.S.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research as well as the APC 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).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset generated by this research, called indo-law data set, has been made available for research purposes at https://github.com/ir-nlp-csui/indo-law accessed on 18 April 2022.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Marzuki, P.M. *Pengantar Ilmu Hukum*; Kencana: Jakarta, Indonesia, 2008; ISBN 978-979-1486-53-8.
2. Schmiegelow, H.; Schmiegelow, M. (Eds.) *Institutional Competition between Common Law and Civil Law: Theory and Policy*, 1st ed.; Springer: Berlin/Heidelberg, Germany, 2014; ISBN 978-3-642-54660-0.
3. Simanjuntak, E. Peran Yurisprudensi Dalam Sistem Hukum Di Indonesia The Roles of Case Law in Indonesian Legal System. *J. Konstitusi* **2019**, *16*, 83–104. [CrossRef]
4. Lotulung, P.E. *Peranan Yurisprudensi Sebagai Sumber Hukum*; Badan Pembinaan Hukum Nasional Departemen Kehakiman: Jakarta, Indonesia, 1997.
5. Butt, S. Judicial Reasoning and Review in the Indonesian Supreme Court. *Asian J. Law Soc.* **2019**, 67–97. [CrossRef]
6. Nuranti, E.Q.; Yulianti, E. Legal Entity Recognition in Indonesian Court Decision Documents Using Bi-LSTM and CRF Approaches. In Proceedings of the 2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 17–18 October 2020; pp. 429–434. [CrossRef]
7. Aletras, N.; Tsarapatsanis, D.; Preoţiuc-Pietro, D.; Lampos, V. Predicting Judicial Decisions of the European Court of Human Rights: A Natural Language Processing Perspective. *PeerJ Comput. Sci.* **2016**, *2*, e93. [CrossRef]
8. Medvedeva, M.; Vols, M.; Wieling, M. Using Machine Learning to Predict Decisions of the European Court of Human Rights. *Artif. Intell. Law* **2019**, *28*, 237–266. [CrossRef]
9. Virtucio, M.B.L.; Aborot, J.A.; Abonita, J.K.C.; Avinante, R.S.; Copino, R.J.B.; Neverida, M.P.; Osiana, V.O.; Peramo, E.C.; Syjuco, J.G.; Tan, G.B.A. Predicting Decisions of the Philippine Supreme Court Using Natural Language Processing and Machine Learning. In Proceedings of the IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 23–27 July 2018; Volume 2, pp. 130–135. [CrossRef]
10. Kowsrihawat, K.; Vateekul, P.; Boonkwan, P. Predicting Judicial Decisions of Criminal Cases from Thai Supreme Court Using Bi-Directional Gru with Attention Mechanism. In Proceedings of the 5th Asian Conference on Defence Technology, ACDT 2018, Hanoi, Vietnam, 25–26 October 2018; pp. 50–55. [CrossRef]
11. Kong, J.; Zhang, L.; Jiang, M.; Liu, T. Incorporating Multi-Level CNN and Attention Mechanism for Chinese Clinical Named Entity Recognition. *J. Biomed. Inform.* **2021**, *116*, 103737. [CrossRef]
12. Li, J.; Jin, K.; Zhou, D.; Kubota, N.; Ju, Z. Attention Mechanism-Based CNN for Facial Expression Recognition. *Neurocomputing* **2020**, *411*, 340–350. [CrossRef]

13. Wijayasingha, L.; Stankovic, J.A. Robustness to Noise for Speech Emotion Classification Using CNNs and Attention Mechanisms. *Smart Health* **2021**, *19*, 100165. [CrossRef]
14. Wu, T.-H.; Kao, B.; Cheung, A.S.Y.; Cheung, M.M.K.; Wang, C.; Chen, Y.; Yuan, G.; Cheng, R. Integrating Domain Knowledge in AI-Assisted Criminal Sentencing of Drug Trafficking Cases. In *Frontiers in Artificial Intelligence and Applications*; Villata, S., Harašta, J., Křemen, P., Eds.; IOS Press: Amsterdam, The Netheralnds, 2020; ISBN 978-1-64368-150-4.
15. Solihin, F.; Budi, I. Recording of Law Enforcement Based on Court Decision Document Using Rule-Based Information Extraction. In Proceedings of the 2018 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2018, Yogyakarta, Indonesia, 27–28 October 2018; pp. 349–354. [CrossRef]
16. Violina, S.; Budi, I. Pengembangan Sistem Ekstraksi Informasi Untuk Dokumen Legal Indonesia: Studi Kasus Dokumen Undang-Undang Republik Indonesia. In *SRITI Proceeding: Seminar Nasional Riset Teknologi Informasi 2009*; SRITI 2009: Yogyakarta, Indonesia, 2009; pp. 135–142.
17. Chen, Y.; Wang, K.; Yang, W.; Qing, Y.; Huang, R.; Chen, P. A Multi-Channel Deep Neural Network for Relation Extraction. *IEEE Access* **2020**, *8*, 13195–13203. [CrossRef]
18. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
19. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; Adaptive Computation and Machine Learning; The MIT Press: Cambridge, MA, USA, 2016; ISBN 978-0-262-03561-3.
20. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1746–1751. [CrossRef]
21. Srinivasamurthy, R.S. Understanding 1D Convolutional Neural Networks Using Multiclass Time-Varying Signalss. Ph.D. Thesis, Clemson University, Clemson, SC, USA, 2018.
22. Kiranyaz, S.; Avci, O.; Abdeljaber, O.; Ince, T.; Gabbouj, M.; Inman, D.J. 1D Convolutional Neural Networks and Applications: A Survey. *Mech. Syst. Signal Process.* **2019**, *151*, 107398. [CrossRef]
23. Indolia, S.; Goswami, A.K.; Mishra, S.P.; Asopa, P. Conceptual Understanding of Convolutional Neural Network—A Deep Learning Approach. *Procedia Comput. Sci.* **2018**, *132*, 679–688. [CrossRef]
24. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. In Proceedings of the 1st International Conference on Learning Representations, ICLR 2013—Workshop Track Proceedings, Scottsdale, AZ, USA, 2–4 May 2013; pp. 1–12.
25. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of Tricks for Efficient Text Classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017—Proceedings of Conference, Valencia, Spain, 3–7 April 2017; Volume 2, pp. 427–431. [CrossRef]
26. Ranzato, M.A.; Boureau, Y.; Cun, Y. Sparse Feature Learning for Deep Belief Networks. In *Proceedings of the Advances in Neural Information Processing Systems*; Platt, J., Koller, D., Singer, Y., Roweis, S., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2007; Volume 20.
27. Gholamalinezhad, H.; Khosravi, H. Pooling Methods in Deep Neural Networks, a Review. *arXiv* **2020**, arXiv:2009.07485.
28. Jeczmionek, E.; Kowalski, P.A. Flattening Layer Pruning in Convolutional Neural Networks. *Symmetry* **2021**, *13*, 1147. [CrossRef]
29. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 7–9 July 2015. [CrossRef]
30. Galassi, A.; Lippi, M.; Torroni, P. Attention in Natural Language Processing. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4291–4308. [CrossRef]
31. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the 31th International Conference on Natural Information Processing System, Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6000–6010.
32. Bahdanau, D.; Cho, K.H.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015—Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015; pp. 1–15.
33. Albon, C. *Machine Learning with Python Cookbook: Practical Solutions from Preprocessing to Deep Learning*, 1st ed.; O'Reilly Media: Sebastopol, CA, USA, 2018; ISBN 978-1-4919-8938-8.
34. Carletta, J. Assessing Agreement on Classification Tasks: The Kappa Statistic. *Comput. Linguist.* **1996**, *22*, 249–254.
35. Manning, C.D.; Raghavan, P.; Schütze, H. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, UK, 2009.
36. Kang, H.; Yang, J. Performance Comparison of Word2vec and FastText Embedding Models. *J. DCS* **2020**, *21*, 1335–1343. [CrossRef]
37. Thavareesan, S.; Mahesan, S. Sentiment Lexicon Expansion Using Word2vec and FastText for Sentiment Prediction in Tamil Texts. In Proceedings of the 2020 Moratuwa Engineering Research Conference (MERCon), Moratuwa, Sri Lanka, 28–30 July 2020; pp. 272–276.
38. Tiun, S.; Mokhtar, U.A.; Bakar, S.H.; Saad, S. Classification of Functional and Non-Functional Requirement in Software Requirement Using Word2vec and Fast Text. *J. Phys. Conf. Ser.* **2020**, *1529*, 042077. [CrossRef]
39. Xian, Y.; Choudhury, S.; He, Y.; Schiele, B.; Akata, Z. Semantic Projection Network for Zero- and Few-Label Semantic Segmentation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8248–8257.

40. Santos, I.; Nedjah, N.; de Macedo Mourelle, L. Sentiment Analysis Using Convolutional Neural Network with FastText Embeddings. In Proceedings of the 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Arequipa, Peru, 8–10 November 2017; pp. 1–5.
41. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. *arXiv* **2013**, arXiv:1310.
42. Grave, E.; Bojanowski, P.; Gupta, P.; Joulin, A.; Mikolov, T. Learning Word Vectors for 157 Languages. *arXiv* **2018**, arXiv:1802.06893.
43. Sokolova, M.; Lapalme, G. A Systematic Analysis of Performance Measures for Classification Tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437. [CrossRef]
44. Holcomb, J.P.; Draper, N.R.; Smith, H.; Rawlings, J.O.; Pantula, S.G.; Dickey, D.A. *Applied Regression Analysis Applied Regression Analysis: A Research Tool*; Springer Texts in Statistics; Springer: New York, NY, USA, 1998; ISBN 978-1-4757-7155-8.
45. Kleinberg, B.; van der Vegt, I.; Mozes, M. Measuring Emotions in the COVID-19 Real World Worry Dataset. *arXiv* **2020**, arXiv:2004.04225.
46. Israeli, O. A Shapley-Based Decomposition of the R-Square of a Linear Regression. *J. Econ. Inequal.* **2007**, *5*, 199–212. [CrossRef]
47. Pishro-Nik, H. *Introduction to Probability, Statistics, and Random Processes*; Kappa Research, LLC: Blue Bell, PA, USA, 2014; ISBN 978-0-9906372-0-2.
48. Yulianti, E.; Kurnia, A.; Adriani, M.; Duto, Y.S. Normalisation of Indonesian-English Code-Mixed Text and Its Effect on Emotion Classification. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 674–685. [CrossRef]
49. Wan, L.; Papageorgiou, G.; Seddon, M.; Bernardoni, M. Long-Length Legal Document Classification. *arXiv* **2019**, arXiv:1912.06905.
50. Berglund, M.; Raiko, T.; Honkala, M.; Kärkkäinen, L.; Vetek, A.; Karhunen, J. Bidirectional Recurrent Neural Networks as Generative Models. In Proceedings of the 28th International Conference on Neural Information Processing Systems—Volume 1; MIT Press: Cambridge, MA, USA, 2015; pp. 856–864.
51. Graves, A. *Supervised Sequence Labelling with Recurrent Neural Networks*; Studies in computational intelligence; Springer: Berlin/Heidelberg, Germany; New York, NY, USA, 2012; ISBN 978-3-642-24796-5.
52. Van Houdt, G.; Mosquera, C.; Nápoles, G. A Review on the Long Short-Term Memory Model. *Artif. Intell. Rev.* **2020**, *53*, 5929–5955. [CrossRef]