JOIV : Int. J. Inform. Visualization, 7(3-2): Empowering the Future: The Role of Information Technology in Building Resilience - November 2023 1041-1047

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

Classification of Industrial Relations Dispute Court Verdict Document with XGBoost and Bidirectional LSTM

Galih Wasis Wicaksono [a,*], Ulfah Nur Oktaviana [a], Said Noor Prasetyo [b], Tiara Intana Sari [a], Nur Putri Hidayah [b], Nur Rohim Yunus [c], Solahudin Al-Fatih [b]

[a] *Informatics Department, Universitas Muhammadiyah Malang, Malang, Jawa Timur, Indonesia*
[b] *Law Department, Universitas Muhammadiyah Malang, Malang, Jawa Timur, Indonesia*
[c] *Faculty of Sharia and Law, Universitas Islam Negeri Syarif Hidayatullah Jakarta, Jakarta, Indonesia*

*Corresponding author: *galih.w.w@umm.ac.id*

***Abstract*—Industrial relations disputes (*Perselisihan Hubungan Industrial* (PHI)) are essential to examine because these disputes represent unbalanced bargaining positions between workers and corporations. On the other hand, there are many PHI documents, so they need to be classified and distinguished from other types of other decisions for other types of civil cases. PHI decisions document can be accessed openly from a special directory of civil courts. This ruling has similarities with other decisions regarding consumer protection or bankruptcy. This study used 450 documents consisting of 255 PHI court decisions and 255 non-PHI court decisions. This study takes the case as a classified part. We use several feature extractions and three methods: Support Vector Machine (SVM), Extreme Gradient Boosting (XGBoost), and Bidirectional Long Short-Term Memory (Bi-LSTM). For SVM and XGBoost classifier, we utilize Frequency-inverse document frequency (TF-IDF). Another classifier needs word embedding Glove Wikipedia Indonesian with a dimension size of 50. Various experiments conducted found that the best classification results used Bi-LSTM with Gloves. This classification has 100% accuracy without overfitting. We found the second result using XGBoost with parameters optimized using random search, while the lowest accuracy results were obtained using the SVM method. The accuracy of the classification results in this study can impact the availability and quality of open legal knowledge that can be utilized by society and for future research.**

***Keywords*— Classification of court documents; bidirectional LSTM; extreme gradient boosting; industrial relations disputes.**

Manuscript received 20 Mar. 2022; revised 16 Sep. 2022; accepted 12 Feb. 2023. Date of publication 30 Nov. 2023.
International Journal on Informatics Visualization is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.

I. INTRODUCTION

Disputes in work fields (industrial relations) seem to be invertible between employers and workers. Such as including rights disputes (e.g., unpaid overtime pays), disputes of interest (examples of non-renewal agreements on the contents of employment agreements), termination disputes, and disputes between trade unions in one company. Industrial relations disputes become important because these disputes occur between workers and corporations/ employers whose bargaining positions are unbalanced.

Industrial relations disputes can be prevented or minimized by providing legal awareness to the public. One source of open legal knowledge that the public can utilize is the court decision on industrial relations disputes. The verdicts of industrial relations cases lie in their classification in a special civil court directory. Therefore, these verdicts bear similarities to other verdicts over consumer protection or insolvency.

This study aims to produce a model of classification of court verdicts in the realm of industrial relations disputes *(Perselisihan Hubungan Industrial/PHI))* with non-industrial relations disputes *(Non-PHI)*. Various studies of the classification of court decision documents were conducted. One of which investigated the classification of the verdict review by comparing the algorithms of the Convolutional Neural Network (CNN), Logistic Regression, Support Vector Machine (SVM), and Random Forest [1]. Another study classified three court verdicts using CNN, Recurrent Neural Network (RNN), and two embedding schemes [2]. The classification of legal verdict documents has proven the effectiveness of using RNN and Long Short-Term Memory (LSTM) [3]. LSTM has also proven to be effective in text processing [4]–[6]. Another comparison found word2vec + CNN embedding combinations to be more accurate than RNN [7].

The implementation of the SVM model [8], [9], Extreme Gradient Boosting (XGBoost) [10], [11], and Bidirectional LSTM (Bi LSTM) [12], [13] document classification also proved successful with varying degrees of accuracy. Therefore, this study uses several machine learning and deep learning models such as SVM, XGBoost, and Bi-LSTM to classify industrial relations disputes and non-industrial relations disputes.

II. Materials and Method

As shown in Figure 1, the research framework used began with literature review research related to the classification of previous court decisions. The next stage involved dataset collection and data preprocessing. The classification phase was carried out in three SVM methods, XGBoost and Bi-LSTM. The final step involved the evaluation and conclusion of classification results.

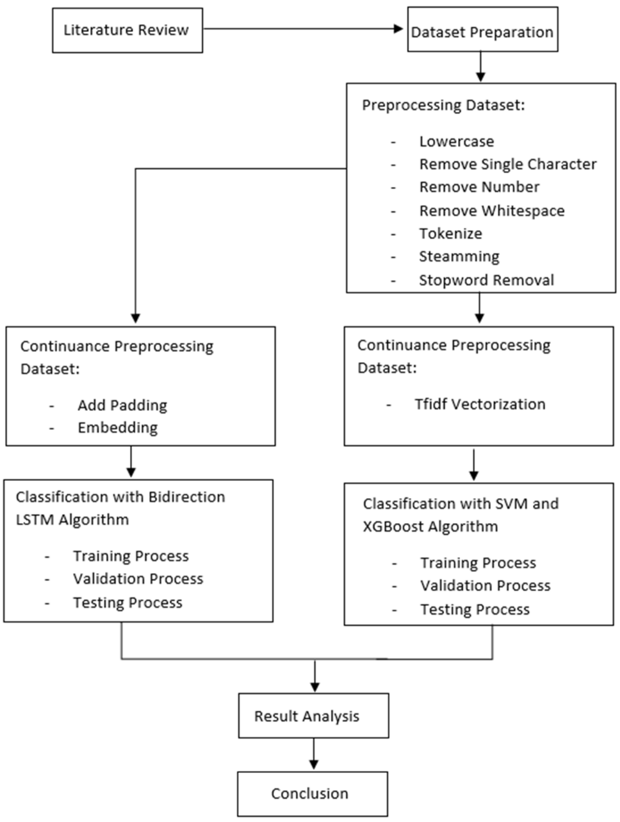

Fig. 1 Research Methodology

A. Data Collection

The data collection process was manually downloading industrial relations dispute court decisions from the supreme court's website in a special directory of civil case decisions. There was also a non-disputed decision on industrial relations to build a balanced dataset of court decisions still contained in the special civil case decision directory—all written in Indonesian.

B. Extraction of Decision Section

The decision of industrial relations disputes consists of several parts: the parties' identity to the dispute, sitting cases, legal considerations, and verdicts. This study did not use all parts of the verdict, but was only restricted to sitting the case. The extraction of the decision was performed automatically using certain keywords in the decision, and the sitting part of the case in the next decision was stored in the *.csv file.

C. Preprocessing

Preprocessing in this study required several stages, namely, (1) Changing the decision document from *.pdf to *.txt; (2) removing information that is not relevant to the classification needs, such as watermarks on decision documents, headers, and footers on each page, page no, etc.; (3) removing single characters; (4) case folding; (5) Removing white space; (6) Eliminating numbers; (7) Tokenizing by using the Natural Language Toolkit (NLTK) library to Indonesian; (8) Steaming using Sastrawi library; 9) Stopwords removal with NLTK library for the Indonesian Language [14], [15]. Up to this stage, preprocessing could be used for all classification stages.

Preprocessing required for the Bi-LSTM method is the addition of padding to make uniform text size with a maximum size of 1000 words. The type of Truncation applied to this study is Post-Sequence Truncation which will cut the length of the sentence at the end of the sentence according to the maximum number of limits. Meanwhile, the Type of Padding used is Post-Padding which adds a zero at the end of the sentence so that the overall length of the sentence will be equal to the number of the longest sentences. An illustration of the use of padding and Truncation can be seen in Fig 2 [16].

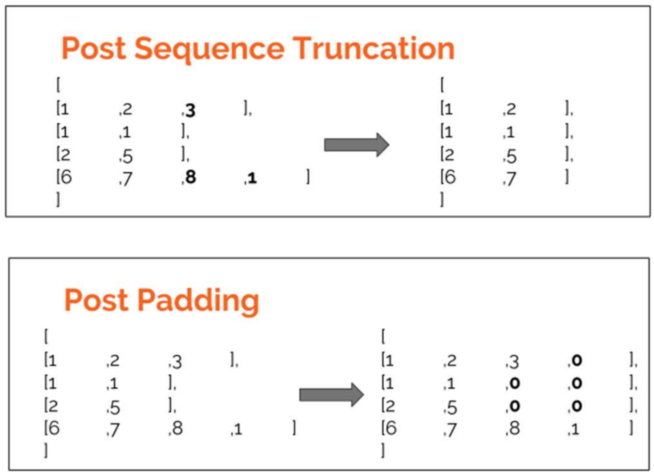

Fig. 2 Post Sequence Truncation and Post Padding

D. Feature Extraction

The feature extraction stage aims to convert words into numbers to be used in the classification process. Feature extraction results in word representations in vector numbers that can be computed using various methods. For the needs of SVM and XGBoost methods, this research used Frequency-inverse document frequency (TF-IDF) to extract features. This method is also used to calculate the weight determination of the importance of each word in the document and the level of similarity [17]. As for mapping words into a vector based on their distribution, Word Embedding Glove Wikipedia Indonesian with a dimension size of 50 was used. The use of word embedding was intended to improve classification performance because the structure in Wikipedia Indonesian can be utilized [18]. This second feature extraction approach was used in the Bi-LSTM method.

E. Classification

This research used the supervised learning method model of support vector machine (SVM) and XGBoost and Deep Learning, namely Bi-LSTM.

1) Support Vector Machine (SVM): SVM will map the training dataset into a high-dimensional feature space. Linear classification using this algorithm will find the boundary between the two classes using a hyperplane. X If the training dataset is N vector $Xi\ (i = 1, 2, \ldots, N)$ of n-dimensional space features $X \epsilon R^n$. Where each vector has a target xi association $yi \epsilon \{-1 + 1\}$. Then it will have an equation function such as Equation (1) where $f(x)$ represents discriminant functions [8].

$$f(x) = w\Phi(x) + b \tag{1}$$

2) XGBoost: XGBoost is a development of the Gradient Boosting Decision Tree, part of machine learning techniques for solving classification and regression problems [19].

3) Bi-LSTM: Bi-LSTM consists of two independent LSTM, which obtain word annotations by summarizing information from two directions and combining sentimental information in annotations. Bi-LSTM equations can be seen in Equation (2), where the number of hidden state forward calculations are based on previous hidden states and vector inputs, while to represent the number of backward states, hidden state calculations are based on the opposite hidden state$fhtxbht$ [20].

$$h_t = [fh_t, bh_t] \tag{2}$$

F. Evaluation

Evaluation of this study used several techniques, such as a confusion matrix, which aims to measure model performance by looking at how much the model can classify correctly, or known as true positive [21]. The second evaluation technique used is the classification report, which checks the model's performance with several parameters such as accuracy, precision, recall, and F1-score [22]. The last evaluation technique used a loss chart to see the Bi-LSTM model's state to determine whether it was in good fit, overfitting, or underfitting. A good fit is when training loss and validation loss are both low values, while overfitting represents a condition where training loss is common. In contrast, validation loss is a high value, and underfitting is a condition where training and validation losses are high value [23].

III. Results and Discussion

The research stages obtained various results with the following details:

A. Dataset

Based on the collection of datasets, we obtained 450 court decisions sourced from the Supreme Court of Indonesia website. An example of a court decision document is shown in Fig 3. The document conversion and labeling process was also performed, followed by balanced data obtained in each class consisting of 255 papers for PHI and 255 Non-PHI decisions.

Direktori Putusan Mahkamah Agung Republik Indonesia
putusan.mahkamahagung.go.id

PUTUSAN
Nomor 7 K/Pdt.Sus-PHI/2020

DEMI KEADILAN BERDASARKAN KETUHANAN YANG MAHA ESA
MAHKAMAH AGUNG

memeriksa perkara perdata khusus perselisihan hubungan industrial dalam tingkat kasasi memutus sebagai berikut dalam perkara antara:

PT KERTA ANGSANA, berkedudukan di Jalan K.H. Wahid Hasyim, Nomor 29, Menteng, Jakarta Pusat, diwakili oleh Dipl. Ing. Heru Setiawan selaku Direktur Utama, dalam hal ini memberi kuasa kepada Winita E. Kusnandar, S.H., MBA., dan kawan-kawan, Para Advokat, beralamat di Equity Tower, Lantai 25, Sudirman Central Business District (SCBD), Lot 9, Jalan Jenderal Sudirman, Kav. 52-53, Jakarta, berdasarkan Surat Kuasa Khusus tanggal 12 Juni 2019;

Pemohon Kasasi/Tergugat;

Lawan

ISANIA GEA, bertempat tinggal di Onowaembo, Desa Onowaembo, Kecamatan Gunung Sitoli Idanoi dan bertempat tinggal di Jalan Salman, Nomor 71, RT. 02, RW. 03, Kelurahan Kebon Jeruk, Jakarta Barat;

Termohon Kasasi/Penggugat;

Mahkamah Agung tersebut;

Membaca surat-surat yang bersangkutan yang merupakan bagian tidak terpisahkan dari putusan ini ;

Menimbang, bahwa berdasarkan surat-surat yang bersangkutan, Penggugat telah mengajukan gugatan di depan persidangan Pengadilan Hubungan Industrial pada Pengadilan Negeri Jakarta Pusat dan memohon kepada Pengadilan untuk memberikan putusan sebagai berikut:

Dalam Pokok Perkara:

1. Mengabulkan gugatan Penggugat untuk seluruhnya;
2. Menyatakan Tergugat telah melakukan perbuatan yang telah bertentangan dengan Undang-Undang Ketenagakerjaan, oleh karenanya

Halaman 1 dari 7 hal. Put. Nomor 7 K/Pdt.Sus-PHI/2020

Fig. 3 Examples of Court Decision

B. Preprocessing Results

The results of preprocessing derived an essential word from each court decision document. Preprocessing results were visualized in a word cloud that displays important words based on the frequency of words appearing in a document. This technique helps analyze important words from multi-documents [24]. In the Non-PHI class, important words were successfully visualized, such as disputes, consumers, businesses, coal sales, please, heavy, and many more, while in the PHI class, important words include work, suction, labor relations, industrial relations, and so forth. Word cloud details are displayed in Fig 4 and Fig 5.

Fig. 4 Word cloud on Non-PHI court decisions

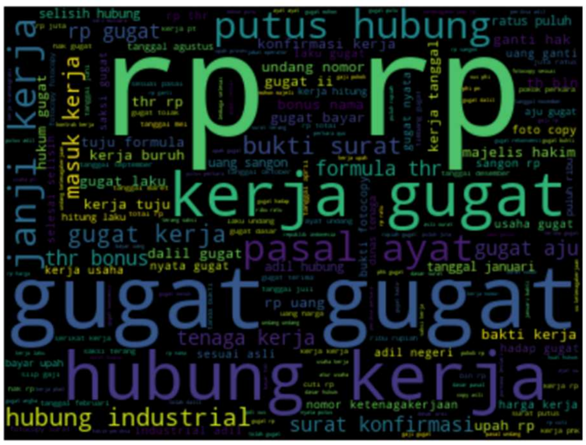

Fig. 5 Word cloud on PHI court decisions

C. Classification Results

1) Support Vector Machine: The classification using SVM with SVC linear kernel function indicated the final accuracy result on the test data accounting for 96%. The results of other parameters, such as precision, recall, and F1-Score, are presented in Table I. From the results of the classification report, both classes have a balanced performance. Other results analysts used a confusion matrix shown from 94 test data; 86 documents were classified precisely, while four were incorrect. The details of the confusion matrix visualization are shown in Fig 6.

TABLE I
SVM CLASSIFIER RESULT

Label	Precision (%)	Recall (%)	F1-Score (%)
PHI	92	100	96
Non-PHI	100	91	95

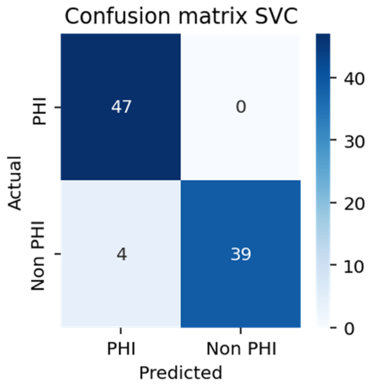

Fig. 6 Confusion Matrix for SVM

2) XGBoost: The classification with XGBoost used two approaches. The first approach using the default parameter obtained the final result accuracy in the test data of 96.6%. XGBoost with default parameters also has a tree architecture, as shown in Fig 7. Classification report results for the first approach are presented in Table II, indicating the results of precision, recall, and F1-Score for PHI and Non-PHI classes. In addition, the analysis of results with a confusion matrix was obtained on the first approach of the 94 test data. This architecture was able to classify 87 documents precisely. In contrast, as many as three papers were misclassified. The details of the confusion matrix results can be observed in Fig 8.

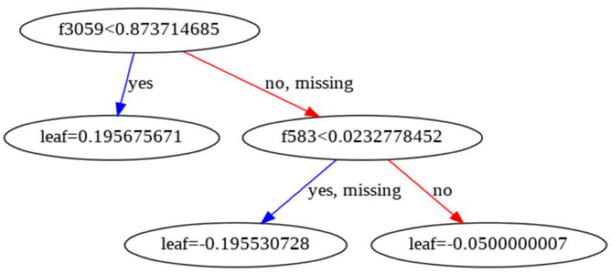

Fig. 7 XGBoost Tree with default parameters

TABLE II
XGBOOST DEFAULT PARAMETER CLASSIFIER RESULT

Label	Precision (%)	Recall (%)	F1-Score (%)
PHI	94	100	97
Non-PHI	100	93	96

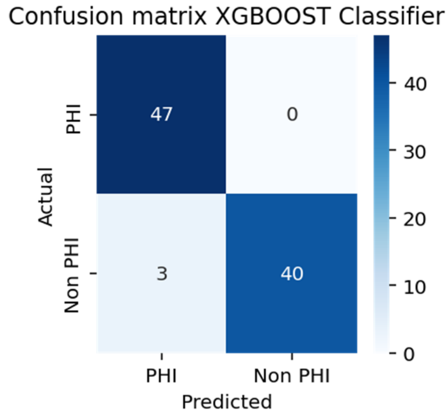

Fig. 8 Confusion Matrix Model XGBoost with Default Parameter

Each machine learning has different parameters that can affect the final performance of the model, and performing Hyperparameter Tuning can be lengthy [25]. Therefore, the second approach aims to optimize the model's performance by using Random search. Random search utilization will look for the best parameter values of some previously defined parameter values.

Four parameters need to be adjusted in the XGBoost model in this study. This study used RandomizedSearchCV provided in the Sklearn library. The defined parameters involved: (1) Learning rate with values: 0.025, 0.05, 0.075, and 0.1; (2) Max Depth with values: 6, 7, and 8; (3) Min child weight with grades: 1, 3, 5, 7, and 9; (4) The last parameter is N Estimator with consecutive values: 1000, 250, and 500. Random search results obtained the best parameters, namely: (1) Learning Rate with a value of 0.05, (2) Max depth of 8, (3) Min child weight of 7, and (4) N Estimator with the amount of 500.

The results of optimizing parameters on XGBoost with random search results in the architecture tree are presented in Fig 9. XGBoost Random Search model performance results compared to the XGBoost architecture default parameters show an increase in accuracy value by 1.1% to 97.7%, while precision value and F1-Score in phi class increased by 2% and 1%. In line with the increase in performance in the results of PHI class classification, non-PHI classes also experienced an increase in recall value and F1-Score by 2% and 2%. The details of classification results with XGBoost Random Search are displayed in Table II and Table III.

Analysts of confusion matrix results, such as presented in Fig 10, showed that this model could classify as many as 88 correct documents and two false documents out of a total of 90 test data. This model is certainly better than the XGBoost model by applying default parameters.

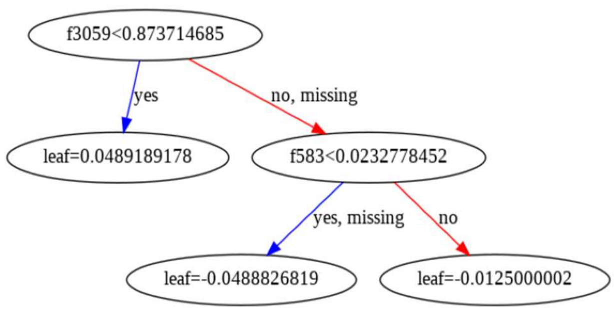

Fig. 9 XGBoost Tree Using Random Search Parameter

TABLE III
XGBOOST RANDOMIZED SEARCH CV RESULT

Label	Precision (%)	Recall (%)	F1-Score (%)
PHI	96	100	98
Non-PHI	100	95	98

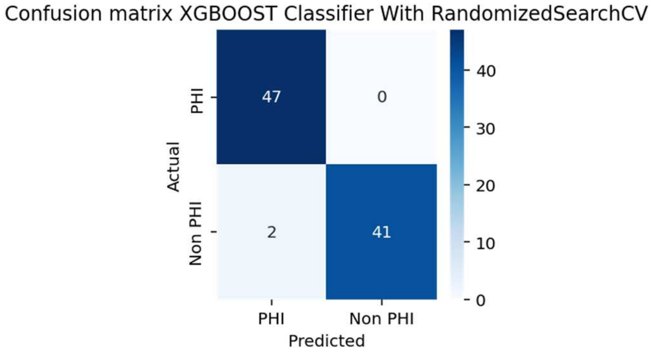

Fig. 10 Confusion Matrix Model XGBoost Random Search

3) Bidirectional LSTM: Bi-LSTM architecture is built with the following layers: (1) Embedding Layer with an embedding size value of 50 according to the number of vectors, (2) Bidirectional Layer LSTM with neurons of 64, (3) Bidirectional Layer LSTM second by 32, (4) Dense Layer with the number of neurons as many as 256, (5) Dropout Layer of 0.6, (6) Dense Layer of 128, (7) Dropout Layer with large size of 0.5, (8) Dense Layer of 32, (9) Dropout Layer with a large 0.5, (10) Output layer with a large one because it is a classification of two classes. The architectural details of this model can be seen in Fig. 11.

The study used neurons of 64 and 32 at the Bidirectional layer. The impact of many neurons on the layer will cause overfitting that affects accuracy performance. The number of neurons at this layer also involves the width of the Bi-LSTM model [26]. In contrast, the use of the dropout layer serves to avoid overfitting in the model by reducing reciprocal information and disconnecting neurons. The model can learn by not being too strict [27], [28].

This Bi-LSTM architecture uses the loss function of binary cross-entropy, the Adam Optimizer because it has a rapid convergence rate [29]. This study's learning rate parameter determining the appropriate model learning speed was 0.0001 [30]. A call-back function was applied to prevent overfitting that monitored the model's performance during training and was carried out at the end of each epoch [31]. The Bi-LSTM model in this study used ModelCheckpoint to store the best model based on loss validation values so that the model with the lowest loss validation value was the final model used.

The final results of the model training gained 100% accuracy on the test data. Table IV shows that the precision, recall, and F1-score results of the PHI and Non-PHI label in the overall test dataset are 100%. Another evaluation can be seen from the loss chart in Fig 12, which shows that the value of loss between training and validation has a low difference. So that this model can be categorized in a good fit condition according to the theory [23]. Meanwhile, if observed based on training graphic and accuracy in Fig. 13, the graph shows the meeting, and there is no significant gap between them.

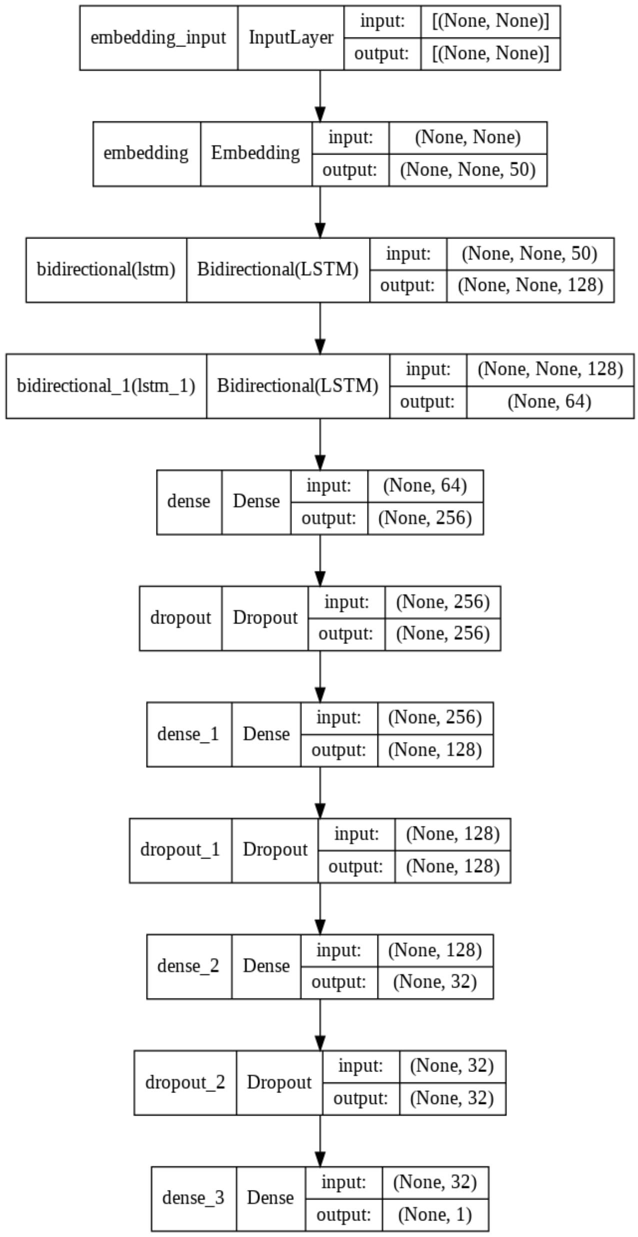

Fig. 11 Bi-LSTM Architecture

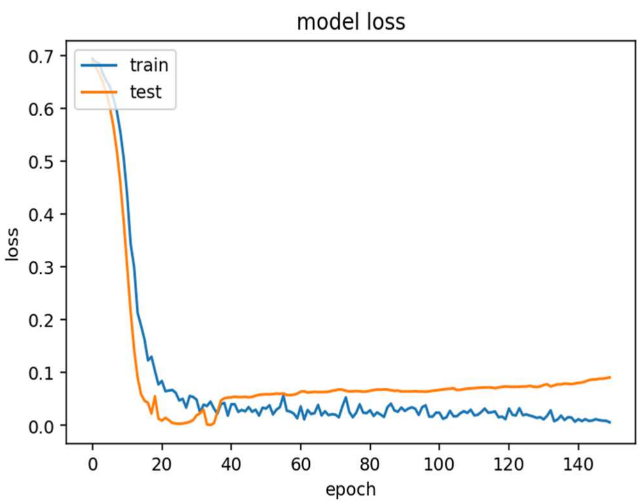

Fig. 12 Model Training Loss Bi-LSTM

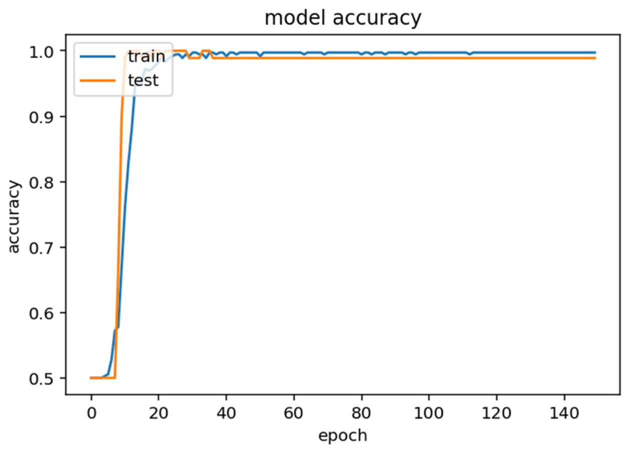

Fig. 13 Model Training Accuracy Bi-LSTM

TABLE IV
BI-LSTM CLASSIFIER RESULT

Label	Precision (%)	Recall (%)	F1-Score (%)
PHI	100	100	100
Non-PHI	100	100	100

The final results of the overall evaluation of the method can be observed in Table V. Bi-LSTM shows the best performing model with an Accuracy value of 100%. Meanwhile, the second-best model XGBoost with a Random Search CV of 97.7%, proved optimal in finding the best parameter value.

TABLE V
THE COMPARISON EXPERIMENT RESULT

Model	The Comparison of model result		
	Accuracy (%)	Precision (%)	Recall (%)
SVM	95.5	96	96
XGBoost	96.6	97	97
XGBoost Random Search CV	97.7	98	98
Bi-LSTM	100	100	100

IV. CONCLUSION

This study's PHI verdict document classification technique used a comparison of SVM, XGBoost, and Bi-LSTM deep learning. The best performances were demonstrated by the Bi-LSTM architecture using the embedding layer and two-layer bi-LSTM. This method can increase accuracy by up to 100%. In addition, the application of several techniques, such as the addition of dropout layers and the use of callbacks, proved effective in avoiding overfitting. The second-best performance uses XGBoost with an accuracy value on test data of 97.7% used Random Search to make it easier to determine four suitable parameters, namely learning Rate, Max Depth, Min child weight, and the N Estimator.

Future research needs to increase the quantity of training and test data, with more classes, to increase the classification depth to subclasses in phi-specific civil case types. In addition, it is also necessary to test the training time and test time so that reliable models are used in various situations and conditions.

REFERENCES

[1] R. Keeling *et al.*, "Empirical Comparisons of CNN with Other Learning Algorithms for Text Classification in Legal Document Review," in *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019*, Dec. 2019, pp. 2038–2042, doi: 10.1109/BigData47090.2019.9006248.

[2] J. Lee and H. Lee, "A Comparison Study on Legal Document Classification Using Deep Neural Networks," in *ICTC 2019 - 10th International Conference on ICT Convergence: ICT Convergence Leading the Autonomous Future*, Oct. 2019, pp. 926–928, doi: 10.1109/ICTC46691.2019.8939926.

[3] M. Y. Noguti, E. Vellasques, and L. S. Oliveira, "Legal Document Classification: An Application to Law Area Prediction of Petitions to Public Prosecution Service," Jul. 2020, doi: 10.1109/IJCNN48605.2020.9207211.

[4] K. Dedes, A. B. P. Utama, A. P. Wibawa, A. N. Afandi, A. N. Handayani, and L. Hernandez, "Neural Machine Translation of Spanish-English Food Recipes Using LSTM," *JOIV Int. J. Informatics Vis.*, vol. 6, no. 2, pp. 290–297, Jun. 2022, doi: 10.30630/JOIV.6.2.804.

[5] Y. Zhang, "Research on text classification method based on lstm neural network model," *Proc. IEEE Asia-Pacific Conf. Image Process. Electron. Comput. IPEC 2021*, pp. 1019–1022, Apr. 2021, doi: 10.1109/IPEC51340.2021.9421225.

[6] R. Saputra, A. Waworuntu, and A. Rusli, "Classification of Indonesian News using LSTM-RNN Method," *Proc. 2021 6th Int. Conf. New Media Stud. CONMEDIA 2021*, pp. 72–77, 2021, doi: 10.1109/CONMEDIA53104.2021.9617187.

[7] S. Undavia, A. Meyers, and J. E. Ortega, "A comparative study of classifying legal documents with neural networks," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018*, 2018, pp. 515–522, doi: 10.15439/2018F227.

[8] M. Goudjil, M. Koudil, M. Bedda, and N. Ghoggali, "A Novel Active Learning Method Using SVM for Text Classification," *Int. J. Autom. Comput.*, vol. 15, no. 3, pp. 290–298, 2018, doi: 10.1007/s11633-015-0912-z.

[9] N. Kalcheva, M. Karova, and I. Penev, "Comparison of the accuracy of SVM kemel functions in text classification," in *Proceedings of the International Conference on Biomedical Innovations and Applications, BIA 2020*, Sep. 2020, pp. 141–145, doi: 10.1109/BIA50171.2020.9244278.

[10] C. A. E. Piter, S. Hadi, and I. N. Yulita, "Multi-Label Classification for Scientific Conference Activities Information Text Using Extreme Gradient Boost (XGBoost) Method," in *2021 International Conference on Artificial Intelligence and Big Data Analytics*, Oct. 2022, pp. 1–5, doi: 10.1109/icaibda53487.2021.9689699.

[11] Z. Qi, "The Text Classification of Theft Crime Based on TF-IDF and XGBoost Model," in *Proceedings of 2020 IEEE International Conference on Artificial Intelligence and Computer Applications, ICAICA 2020*, Jun. 2020, pp. 1241–1246, doi: 10.1109/ICAICA50127.2020.9182555.

[12] R. Anhar, T. B. Adji, and N. Akhmad Setiawan, "Question classification on question-answer system using bidirectional-LSTM," Jul. 2019, doi: 10.1109/ICST47872.2019.9166190.

[13] J. Li, Y. Xu, and H. Shi, "Bidirectional LSTM with Hierarchical Attention for Text Classification," in *Proceedings of 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference, IAEAC 2019*, Dec. 2019, pp. 456–459, doi: 10.1109/IAEAC47372.2019.8997969.

[14] F. Hartono, R. Lim, and L. P. Dewi, "Pembuatan Sistem Rumah Pintar dengan Voice Assistant di Raspberry Pi," *J. Infra*, vol. 8, no. 1, pp. 82–88, Apr. 2020.

[15] P. Verma, A. Goyal, and Y. Gigras, "Email phishing: text classification using natural language processing," *Comput. Sci. Inf. Technol.*, vol. 1, no. 1, pp. 1–12, 2020, doi: 10.11591/csit.v1i1.p1-12.

[16] M. Dwarampudi and N. V. S. Reddy, "Effects of padding on LSTMs and CNNs," 2019.

[17] J. Dr. Menyhárt and J. H. Gomes Da Costa Cavalcanti, "LSI with Support Vector Machine for Text Categorization – a practical example with Python," *Int. J. Eng. Manag. Sci.*, vol. 6, no. 3, pp. 18–29, 2021, doi: 10.21791/ijems.2021.3.2.

[18] D. & E. A. A. Sudana, *Seminar Tahunan Linguistik 2018*, no. Setali. 2016.

[19] S. Thongsuwan, S. Jaiyen, A. Padcharoen, and P. Agarwal, "ConvXGB: A new deep learning model for classification problems based on CNN and XGBoost," *Nucl. Eng. Technol.*, vol. 53, no. 2, pp. 522–531, 2021, doi: 10.1016/j.net.2020.04.008.

[20] C. W. Chen, S. P. Tseng, T. W. Kuan, and J. F. Wang, "Outpatient text classification using attention-based bidirectional LSTM for robot-assisted servicing in hospital," *Inf.*, vol. 11, no. 2, 2020,

doi: 10.3390/info11020106.

[21] K. Shah, H. Patel, D. Sanghvi, and M. Shah, "A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification," *Augment. Hum. Res.*, vol. 5, no. 1, 2020, doi: 10.1007/s41133-020-00032-0.

[22] D. M. W. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," pp. 37–63, 2020.

[23] R. Doon, T. Kumar Rawat, and S. Gautam, "Cifar-10 classification using deep convolutional neural network," in *1st International Conference on Data Science and Analytics, PuneCon 2018 - Proceedings*, 2018, no. x, pp. 1–5, doi: 10.1109/PUNECON.2018.8745428.

[24] Chandrapaul, R. Soni, S. Sharma, H. Fagna, and S. Mittal, "News analysis using word cloud," in *Lecture Notes in Electrical Engineering*, 2019, vol. 526, pp. 55–64, doi: 10.1007/978-981-13-2553-3_6.

[25] A. Haidar, B. Verma, and R. Haidar, "A Swarm based Optimization of the XGBoost Parameters," vol. 16, no. 4, pp. 74–81.

[26] C. Bian, H. He, and S. Yang, "Stacked bidirectional long short-term memory networks for state-of-charge estimation of lithium-ion batteries," *Energy*, vol. 191, p. 116538, 2020, doi: 10.1016/j.energy.2019.116538.

[27] T. Jiang, D. Wang, L. Sun, H. Yang, Z. Zhao, and F. Zhuang, "LightXML: Transformer with Dynamic Negative Sampling for High-Performance Extreme Multi-label Text Classification," 2021.

[28] G. Chen, P. Chen, Y. Shi, C.-Y. Hsieh, B. Liao, and S. Zhang, "Rethinking the Usage of Batch Normalization and Dropout in the Training of Deep Neural Networks," 2019.

[29] B. Škrlj, J. Kralj, N. Lavrač, and S. Pollak, "Towards Robust Text Classification with Semantics-Aware Recurrent Neural Architecture," *Mach. Learn. Knowl. Extr.*, vol. 1, no. 2, pp. 575–589, 2019, doi: 10.3390/make1020034.

[30] S. Sun, Z. Cao, H. Zhu, and J. Zhao, "A Survey of Optimization Methods from a Machine Learning Perspective," *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3668–3681, 2020, doi: 10.1109/TCYB.2019.2950779.

[31] Z. Yang *et al.*, "TextBrewer: An Open-Source Knowledge Distillation Toolkit for Natural Language Processing," 2020, pp. 9–16, doi: 10.18653/v1/2020.acl-demos.2.