PAPER

A Two-Stage Classifier That Identifies Charge and Punishment under Criminal Law of Civil Law System

Sotarat THAMMABOOSADEE[†a)], ***Member***, **Bunthit WATANAPA**[††b)], **Jonathan H. CHAN**[††c)], ***and*** **Udom SILPARCHA**[††d)], ***Nonmembers***

SUMMARY A two-stage classifier is proposed that identifies criminal charges and a range of punishments given a set of case facts and attributes. Our supervised-learning model focuses only on the offences against life and body section of the criminal law code of Thailand. The first stage identifies a set of diagnostic issues from the case facts using a set of artificial neural networks (ANNs) modularized in hierarchical order. The second stage extracts a set of legal elements from the diagnostic issues by employing a set of C4.5 decision tree classifiers. These linked modular networks of ANNs and decision trees form an effective system in terms of determining power and the ability to trace or infer the relevant legal reasoning behind the determination. Isolated and system-integrated experiments are conducted to measure the performance of the proposed system. The overall accuracy of the integrated system can exceed 90%. An actual case is also demonstrated to show the effectiveness of the proposed system.

key words: *criminal law, data mining, decision tree, legal reasoning, neural network*

1. Introduction

The main contribution of this study is to provide a basis for understanding the sequential process of legal reasoning, and enabling a forecast to be made of the final charges and the theoretical punishment ranges under the criminal law code of the Civil Law system of Thailand. Only criminal cases in the offences against life and body section (article 288-297) are considered at this stage of research because of the wide availability of recorded cases. The result of this study is by no means a substitute for the prestigious and sophisticated legal system of Thailand.

Civil Law is one of the two most common legal systems. The main distinction between Civil Law and Common Law is that, under the Common Law system, the law is unwritten. Prior decisions can form strict precedents for subsequent judgments of the court. On the other hand, any changes to the written legal codes in the Civil Law system must be conducted through the legislative system; thus prior decisions do not form a rule or principle for future cases but rather simply offer recommendations [1]. Note that the term "Civil Law" is widely used in 2 contexts. First, it implies a legal system as stated. Another implication is to a set of codes that are intended to resolve the conflicts among individuals or organizations. Nonetheless, the latter implication is not relevant to this paper.

By the legal process of the Civil Law system, criminal facts can be collected, investigated, and collated into a more abstract layer that conforms to the elements and frameworks in the law articles. To reflect this nature, Thammaboosadee et al. [2] proposed sequential data analysis for the judicial process as follows: (i) criminal facts are collected and formalized into fact-level data, (ii) fact-level data are classified into diagnostic issues (case-level data), and, lastly, (iii) case-level data are analyzed and classified into legal elements (legal-level data), for correspondingly applicable articles for identifying charges and the range of possible sentences. This systematic inferencing process yields decisions that can be defended by the system [3].

To obtain the diagnostic issues from case facts, a modular artficial neural network (ANN) [4] was developed so that its modular architecture imitates the back-end processes of investigation and analysis. For the indentification of the legal elements, the C4.5 decision tree [5] was selected to construct the model. Its inductive ability is highly beneficial for interpreting legal reasons to human users. This integrated system is an extension of the work [6] which fully focused on the determination of diagnostic issues. Readers can refer to this paper to obtain extensive information about the detailed data treatment of the first classifier of the two-stage classifier proposed in this study.

The remainder of the paper is organized as follows: the next section discusses relevant theories in the law domain, as well as some backgrounds to the selected classification methods and related past works in the legal domain. Section 3 introduces the selected methodologies used in the classifying process, including the architectural design, preprocessing methods, and details of the experimental design. Section 4 contains the experimental results and a discussion of technical issues as well as some insights into the legal domain. Finally, we conclude the paper and suggest directions for future research.

2. Background and Related Works

To systematize the legal reasoning problem, a decision tree, such as C4.5 and its sibling, is well known as a powerful

Manuscript received June 7, 2013.
Manuscript revised November 2, 2013.

[†]The author is with the Faculty of Engineering, Mahidol University, Thailand.
[††]The authors are with the School of Information Technology, King Mongkut's University of Technology Thonburi, Thailand.
a) E-mail: zotarat@gmail.com
b) E-mail: bunthit@sit.kmutt.ac.th
c) E-mail: jonathan@sit.kmutt.ac.th
d) E-mail: udom@sit.kmutt.ac.th

DOI: 10.1587/transinf.E97.D.864

interpretable computational method. C4.5, an extension of the ID3 algorithm, is an algorithm used to generate a decision tree whose construction is based on the concept of information entropy [7]. First, the algorithm finds an effective split of the data based on the highest normalized information gain [7] for each attribute. It then creates a decision node using the selected node and the expected value of splitting. The algorithm recurs on the sublist of the obtained splitting on the selected attribute and adds these nodes as child nodes. Equation (1) shows the calculation of the information gain of an attribute a for a set of cases T when a is discrete and $T_1, T_2, \ldots, T_s$ are the subsets of T consisting of cases with distinct known values for attribute a. The information gain is based on the entropy function, as shown in Eq. (2). The function $freq(C_j, T)$ calculates the frequency that case C is classified as T_j. C4.5 uses on the information gain ratio to determine the split of data, as shown in Eq. (3), where P is the probability function.

$$gain(a) = info(T) - \sum_{i=1}^{s} \frac{|T_i|}{|T|} \times info(T_i) \tag{1}$$

$$info(T) = -\sum_{j=1}^{Nclass} \frac{freq(C_j, T)}{|T|} \times log_2(\frac{freq(C_j, T)}{|T|}) \tag{2}$$

$$Split(T) = -\sum_{j=1}^{s} \frac{|T_j|}{|T|} \times log_2(P(\frac{|T_j|}{|T|})) \tag{3}$$

Unfortunately, the C4.5 decision tree usually does not provide the best prediction accuracy for classifications although it effectively provides insights into the interaction between variables [8]. Gutierrez and Leroy [9] investigated the C4.5 decision tree for use in a crime prediction system that aims to find and predict unreported crime. The system was experimented and compared in terms of accuracy and insights into the attributes of the tree. The accuracy of these models is not satisfactory, considering with an its achievement of only 56-75%. This challenges the introduction of a more sophisticated system.

A meta-heuristic method such as an ANN could be an effective solution, especially when the domain is large and complex. Typically, an ANN is a computational model that is inspired by the structural and functional aspects of the biological neural network system. It consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach. There are several types of ANN. In this research, an ANN with a feedforward and back-propagation architecture [10] is chosen. In principle, this type of ANN iteratively obtains a specified set of inputs and feeds the learned importance of each of them (synapses coded into weights) through an organized network of neurons in such a way that errors of the network (gaps between the known target and calculated output) can be propagated back to the upstream nodes of the neural net for tuning of the synapses.

An example of a study that applied a conventional ANN to legal tasks was proposed by Oatley et al. [11]. In their study, evidence was applied to forensic science using data from various sources to discover the criminal factors that determined the type of crime. A simple feedforward back-propagation ANN architecture was applied. Although the lack of an inferable structure such as modularity or tree limited the effectiveness of this application to a detection system, it demonstrated that an ANN is an effective tool with an accuracy range of 75-92% for determining legal factors.

Another example was presented by Theresa and Raj [12]. They reported the effective use of backpropagation ANN in the classification of murder cases. The classification results were compared with the decision of an expert yielding an accuracy of 96.67%. Although this result is satisfactory, interpretability is lacking.

Specific to the level of investigation, for the detection of credit card fraud, which is a legal task carried out at the police level, Aleskerov et al. [13] reported another successful application of an ANN. They developed a data integration methodology for their system that involved various data transformation techniques. The neural network architecture of their system was a fully connected feedforward network with 3 fixed layers. The system can classify cases of fraud into 4 groups. A detection rate of 85% was reported with no reasoning ability.

Since the conventional ANN works as a black box that virtually imitates the interaction of the neural system in a human brain [14], its obtained results yield no interpretation. To eliminate this limitation, in this paper we propose a modular ANN architecture that is heuristically designed into a multi-echelon network of specialized identifiers. Once a result is obtained from the system, one can trace back the echelon to derive the interpretation. This type of ANN architecture has also been successfully applied in some other domains. For example, the environmental impact of ICT and e-business [15], the emotion-based classification of movie clips [16], and English pronunciation reasoning and protein prediction [17].

A successful application of a modular ANN in the law domain was proposed by Stranieri and Zeleznikow [18]. They developed the Split-Up system which determines the final judgment in cases of Australia's family law domain, specifically the distribution of marital property in divorce cases. Various artificial intelligence and knowledge discovery techniques were applied. Hierarchical modules of legal factors were preconstructed to imitating the hierarchical decision making process of the court. The authors assumed that the judge in the Australian family court in each specific case makes a decision by referring to the diagnostic issues and verdicts in previous cases. Reflecting this nature, a modular ANN was applied to identify relevant factors for considering cases. The complete system was put online and interacted with users with the help of a decision tree [19]. However, this work is different from the domain of our work. The required final results in the Civil Law system, especially in criminal law code, should specify all important legal elements based on the legal body and map onto charges.

In the next section, we explore how the 2 key methodologies of C4.5 and ANN contribute to the holistic design and evaluation plan. Other methods of the the integral parts, such as preprocessing techniques, are also discussed to establish an integrative view of the architecture and organization of the system.

3. Methodology

To construct a classification system to forecast legal charges based on a huge set of fact-level attributes in respect of the above-mentioned requirements, a proposed two-stage classifier is applied. The benefit of this is at least two fold: a reduction in the training data constraint and conformance to the legal decision process in the real world. [2] The first stage is to identify the case-level attributes based on obtained fact-level attributes. This is very important and requires computing schemes that can comprehend the subtleties way of human judgment. This classifier level is built with an ANN algorithm equipped with a modular structure. The second stage is to identify the legal-level attribute based on the case-level attributes acquired in the first stage. The C4.5 decision tree method is used because of its inferable results. Finally, the identified legal elements are then mapped onto the legal charge codes, which provide the sentences and ranges of punishments. Figure 1 shows the architectural design of our proposed system.

As shown in Fig. 1, not all of the fact-level attributes are fed into the case-level classifier. Uncomplicated attributes can be directly mapped using explicit rules. In the same fashion, only a certain number of case-level attributes are required in decision tree analysis. The numbers of attributes acquired are 99, 25, and 9 for the fact level, case level and legal level respectively, as summarized in Table 1.

Owing to the high dimensionality of the system, most of the attributes including the uncomplicated fact-level attributes have to be preprocessed [20], for example, the age of an offender can be in 2 different format: numerical or textual.

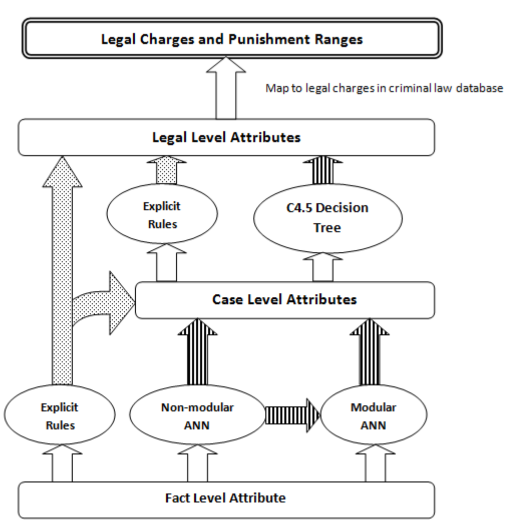


Fig. 1 Overall data layers and identification methods.

In our preliminary study of selecting methods to classify case-level attributes, we first investigated with the traditional classification methods of the C4.5 decision tree and the Naive Bayes classifier because their easy-to-interpret results. However, the empirical results had an unsatisfactory accuracy of 65% to 77%. This implies the existence of complexity in classifying the case-level attributes, leading us to consider more refined techniques such as an ANN. Note that the key features of interpretability and understandability of this intermediate decision can be fulfilled by applying modularity in an ANN. It is the front-end subsystem of the later stage of legal-level identification that requires the C4.5 algorithm to provide the inference engine of artificial human reasoning for legal-based arguability. Other sophisticated human-interpretable classification methods should be deployed to achieve more reliable results in future work.

3.1 Data-Preprocessing Procedures

Initially, the attributes in all levels are extracted from the verdicts stored in TCXML [21]. The TCXML (Thai Court XML) is an XML (Extensible Markup Language) [22] data model that was extended from the Global Justice XML data model [23] proposed to collect the details of the verdicts in incident facts, the diagnostic issues, and the final judgments made by the courts, especially in the cases in Thailand. Figure 2 shows a schematic of the main elements of TCXML to illustrate how it is structurized. The verdict usually lists the involved charges at the top part of the document, which refers to the header tag of TCXML. The described incidents of each case are stored in both the "VerdictComplainantStatementList" and "VerdictCourtEventList" tags. The incidents appeared in the former tag refer to the facts in the plaintiff's accusation at the police and prosecutor levels. The incidents in the latter tag refer to the judge's conclusion, which usually describes the proven facts, the diagnostic issues, and the final judgment.

In our system, the extracted TCXML elements contain all of the required data levels and the related law articles. We set the scope of the legal domain to the offence against life and body section of the criminal law code. The verdicts

Table 1 Summary of attributes acquired on each data layer.

Data layer	Explicit rules	Data analysis	Total
Fact-Level	-	-	99
Case-Level	12	13	25
Legal-Level	1	8	9

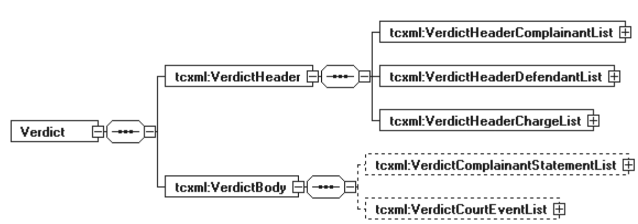


Fig. 2 Main elements of TCXML structure.

of 150 cases from the Supreme Court in the period of 1990-2010 were randomly selected from a total of 4078 available verdicts as a data set. The limited amount of available data may affects the reliability of the system through possible undertraining, although the ANN and decision tree do not have strict requirements of input size [24]. To preprocess the input data while reducing the dimensionality, 2 schemes are introduced as discussed below.

3.1.1 Attribute Filtering by Principal Component Analysis (PCA)

The objective of PCA [25] is to reduce the dimensionality (number of variables) of a dataset while retaining most of the original variability of the data. The first principal component takes into accounts as much of the variability in the data as possible, and each successive component account for as much of the remaining variability as possible. In brief, PCA generates a matrix of covariance from all inputs. Then this matrix is used to calculate the eigenvectors and eigenvalues. The eigenvectors (features) with a low eigenvalue are eliminated from the model.

3.1.2 Attribute Ranking by Support Vector Machine (SVM)

The SVM classification algorithm [26], originally proposed to solve two-class problems, finds a separation between hyperplanes defined by classes of data. Its goal is to measure the margin of separation of the data rather than to match features (or attributes); thus, the SVM algorithm can avoid falling into the trap of local optimality and operate effectively even with moderately large feature sets. To enhance the SVM classification algorithm as a feature-ranking method, Guyon et al. [27] proposed a new integrated method of SVM classification and proposed a correlation based feature-ranking method called recursive feature elimination (RFE) [28]. This new integrated method, named SVM-RFE, is selected in this paper because of its good performance in attributes subset searching, especially for complex data. The algorithm of the SVM-RFE attribute ranking method which is shown in Fig. 3.

Algorithm of SVM-RFE attribute ranking method

Input:

Training examples $X_0 = [x_1, x_2, ..., x_k]$

Class labels $y = [y_1, y_2, ... , y_k]$

Output: Feature ranked list r.

s = [1, 2,... n] //Initialize subset of surviving features

r = [ ] //Initialize feature ranked list

repeat

X = restrict(X_0,s) //Restrict training examples to good feature indices.

a = SVM(X, y) //Train the classifier.

w = Weight(a_k,y_k,x_k) //Compute the weight vector of dimension length.

$c_i = (w_i)^2$, for all i //Compute the ranking criteria.

f = min(c) //Find the feature with the smallest ranking criterion.

r = update(s(f), r) //Update feature ranked list.

s = elim(s,f) // Eliminate the feature with the smallest ranking criterion.

until s = [] //All features are ranked.

Fig. 3 SVM-RFE attribute ranking algorithm.

Compared with PCA, representing which is a well-known statistical technique, SVM-RFE, which is a classification-based method (building a supportive model to reduce dimensionality), should be more systematic and context-sensitive. Theoretically, the model-based method should be a more accurate classification model but have higher computational and memory complexities [29]. The accuracy of these two methods was considered in our experiment.

Next, we discuss the model settings in the experiments.

3.2 Classification System

The proposed classification system consists of 2 main engines that work in sequence, one to identify the diagnostic issues and one to identify the legal elements.

3.2.1 Identification of Diagnostic Issue

The selected classifier in this stage is an ANN.

In our ANN model, the prospective extracted inputs are 99 raw attributes. According to Witten and Frank [20], an effective data-preprocessing method is required to treat such a huge volume of data. This high-dimensional dataset is treated by PCA to ensure that only significantly correlated inputs are taken into account. Considering the target classes, the targeted case-level attributes are categorized into 25 groups. Tweleve out of the 25 case-level attributes are explicitly identifiable by certain simple rules with only 1 or 2 attributes, so only 13 classes are determined by the ANN. Tables 2 and 3 respectively show the definitions and characteristics of the case-level attributes in both of the ANN (A) analysis and in the idenfication of explicit rules (R). An example of a nonmodular neural network is shown in Fig. 4. The example considered here is the Result_Severity class (network C9) which consists of high-dimensional of inputs and would need reconstruction as a modularity. In Fig. 4, the conventional ANN model feeds all 16 inputs of fact-level attributes (I1-I16) through eight hidden nodes (H1-H8) and gives the outcome as the level of force in six of the output nodes (O1-O6).

The modular structure in the ANN emulates the divide-and-conquer method so that the final outcome of the system is consolidated from the other ANN modules [4]. Figure 5 illustrates the relationships among 13 ANN modules. For example, in this modular structure, ANN module 3 is dependent on the outcome of ANN module 9, which in turn requires the output of modules 8, 10, and 13. The designed structure reflects hypothetical induction rules in the reasoning process of the judicial system. Heuristically, each classification module has a designated task to classify diagnostic issues using different attribute sets. Some attributes may appear in many sets and some modules may rely on the results of other modules. The model can be separated into two types: independent modules (dashed shapes) and dependent modules (solid shapes).

Table 2 Definitions of all case-level attributes.

Net	Name	Description
C1	Act_ForceLevel	The level of the force applied by the offender.
C2	Act_HasChanceToConsider	Whether the action was immediately performed.
C3	Act_IsCruel	Whether the action was cruelly performed to the victim.
C4	Act_Necessity	The level of necessity of the offender in the act.
C5	Act_Plan	Whether the action was planned it was performed.
C6	Conceal_Cruel	Whether method of the concealment after the action was cruel.
C7	Motivation_SevereReason	The severity of the reason that made the offender perform the incident
C8	Organ_Importance	The importance of the affected organs
C9	Result_Severity	The severity of the effect of the incident on the victim
C10	Sore_Severity	The severity of the victim's wound.
C11	TrafficAct_MustDriveCarefully	In traffic incidents, whether the offender should have been driving more carefully.
C12	TrafficAct_UnableToStopInTime	In traffic incidents, whether the incident occurred suddenly
C13	Weapon_Severity	The lethality of the weapon used in the incident
C14	Act_AbleToKillButDidnt	Whether the offender had a chance to kill the victim but restrained.
C15	Act_Foreseeable	Whether the incident could be foreseen by the offender before it was performed.
C16	Act_Direct	Whether the incident affected the victim directly.
C17	Act_MistakeInPerson	Whether the action was performed on the wrong person.
C18	Act_TortureBeforeDie	Whether the victim was tortured before he/she died.
C19	Location_LimitedLocation	Whether the location was limited by the construction structure.
C20	Location_OviousLocation	Whether the location was obvious enough to perform an action.
C21	Motivation_MotivationSeverity	The severity of the reason that caused the action.
C22	Motivation_AngryLevel	Whether the offender was angered before the action was performed.
C23	Result_Causation	The causation between the action and the result.
C24	Time_ContinuouslyTimePeriod	Whether the action was continuous from the start to end.
C25	Weapon_IsStandard	Whether the weapon used was made in a standard condition.

3.2.2 Identification of Legal Element

The role of this classifier is to determine the legal level attributes based on the identified case-level attributes obtained

Table 3 Characteristics of all case-level attributes.

Net	ANN(A)/ Rules(R)	No. of Input Attributes	Output Category
C1	A	21	None: Low: IfHighThenInjure: IfHighThenDie: High
C1	A	21	None: Low: IfHighThenInjure: IfHighThenDie: High
C2	A	27	No: Yes
C3	A	20	No: Yes
C4	A	23	None: Low: Necessity: Over
C5	A	28	No: Yes
C6	A	18	No: Yes
C7	A	5	No: Yes
C8	A	3	No: Yes
C9	A	16	None: Low: MindSevere: PhysicalSevere: High: Death
C10	A	4	No: Yes
C11	A	6	No: Yes
C12	A	6	No: Yes
C13	A	5	No: Low: High
C14	R	-	No: Yes
C15	R	-	No: Low: Danger: Death
C16	R	-	No: Yes
C17	R	-	No: Yes
C18	R	-	No: Yes
C19	R	-	No: Yes
C20	R	-	No: Yes
C21	R	-	No: Yes
C22	R	-	No: Yes
C23	R	-	No: Yes
C24	R	-	No: Yes
C25	R	-	No: Yes

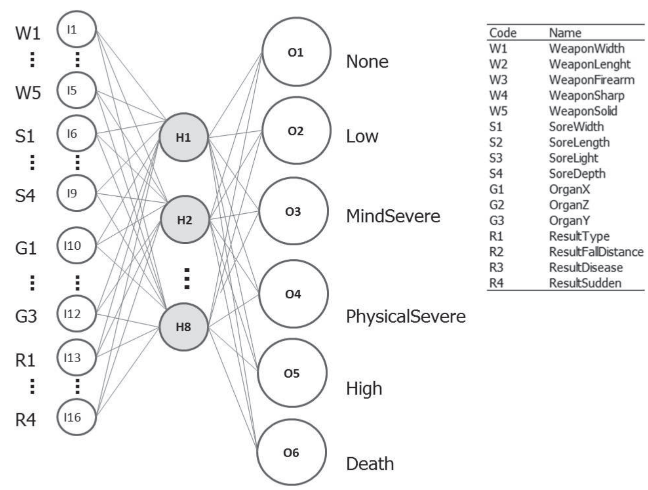


Fig. 4 Network topology for Result_Severity (C9) in nonmodular structure.

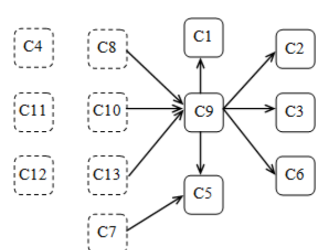


Fig. 5 Modular structure of the identification system for diagnostic issue.

Table 4 definition and values of all legal-level attributes.

Name (Abbreviation)	Code	Description
Act_Intention_Result (ACT)	ACT10	PettyHarm-Intention-Noresult
	ACT20	Anyaction-Negligence-Severe
	ACT45	Harm-Intention-Attempt
	ACT50	Harm-Intention-NotSevere
	ACT60	Anyaction-Negligence-Death
	ACT65	Harm-Intention-Severe
	ACT70	Harm-Intention-Death
	ACT95	Homicide-Intention-Attempt
	ACT100	Homicide-Intention-Death
Anger (ANG)	ANG0	No anger
	ANG100	Anger
Awareness (AWA)	AWA0	No awareness
	AWA50	Moderately aware
	AWA100	Fully aware
Concealment (CON)	CON0	No concealment
	CON100	Concealment
Cruelty (CRU)	CRU0	No Cruelty
	CRU100	Cruelty
Plan (PLA)	PLA0	No Plan
	PLA100	Plan
Prevention (PRE)	PRE0	No prevention
	PRE42	Prevent from harm
	PRE50	Necessary prevention
	PRE100	Over excess prevention
Unachievable Attempt (UNA)	UNA0	None
	UNA100	Unachievable attempts
OffenderAge (AGE)	OFF10	Age of offender is below 10 years
	OFF15	Age of offender is below 15 years
	OFF18	Age of offender is below 18 years
	OFF20	Age of offender is below 20 years
	OFF25	Adult

from the modular ANN model. The possible legal elements are forecasted and then mapped to charges. Also, the interpretation of why specific articles are chosen has to be inductively arguable based on the specified set of diagnostic issues. At this level, an inductive inference engine is needed to emulate the reasoning process. In our system, the C4.5 decision tree [5] is selected as the classifier.

The legal-level attributes are defined on the basis of the classes derived from the criminal law code ontology [30], which are constructed on the basis of criminal law theory and elements in the law code [31]. For the purpose of testing, the scope of the structuralized legal-level attributes is chosen to be the law articles that appear in the offences against life and body section of Thailand's criminal law. Tables 4 and 5 respectively show the definition with values of each legal-level attribute and their related law articles or change effects. The letter "p" after the article number refers

Table 5 The related law article and change effect of all legal-level attributes.

Name	Code	Law article no.	Change Effect
ACT	ACT10	391	
	ACT20	300	
	ACT45	295,80	
	ACT50	295	
	ACT60	291	
	ACT65	297	
	ACT70	290	
	ACT95	288,80	
	ACT100	288	
ANG	ANG0	-	
	ANG100	72	
AWA	AWA0	65p1	
	AWA50	65p2	
	AWA100	-	
CON	CON0	-	
	CON100		288 to 289(7) 295 to 296 290 to 290p2
CRU	CRU0	-	
	CRU100		288 to 289(5) 295 to 296 290 to 290p2
PLA	PLA0	-	
	PLA100		288 to 289(4) 295 to 296 290 to 290p2
PRE	PRE0	-	
	PRE42	67	
	PRE50	68	
	PRE100	69	
UNA	UNA0	-	
	UNA100	81	
AGE	OFF10	73	
	OFF15	74	
	OFF18	75	
	OFF20	76	
	OFF25	-	

to the paragraph number within the article. For example "65p2" refers to the second paragraph of article no.65. Note that the legal-level attribute OffenderAge (AGE) is the only one that is identifiable by explicit rules.

According to Table 4, there are 8 legal-level attributes treated as target classes of the classifier. The first one, "Act_Intention_Result" (ACT), has the largest dimension with 10 different possible values. Each value represents a different combination of 4 legal attributes: act, intention, commitment and result. This combination of attributes achieves synergy because individual values are ambiguous but aggregation gives a contextual and bonded value. As an example, let us assume that an offender slashed a victim, resulting in minor injury. The contingency could be that, in some cases, the act of the offender may be judged as an attempt at homicide with intention (ACT95) if the injury affected the victim's head. Conversely, the offender's act may be judged as successful intentional harm (ACT65) if the victim's leg was injured.

It should be clarified here clarified here that the first paragraph of Article 59 of the criminal codes states that

"*A person shall be criminally liable only when such person commits an act intentionally, except in case of the law provides that such person must be liable when such person commits an act by negligence, or except in case of the law clearly provides that such person must be liable even though such person commits an act unintentionally.*" This means, in the charges identification process, ACT is the key element to be identified since it has been defined as a combination of the type of action, the type of intention, the level of commitment, and the result. Logically, if the incident does not satisfy any defined offence legal element set, the other classes need not be identified. The flow of charge identification process is summarized in Fig. 6.

The value of each legal element should be determined from a set of diagnostic issues by an inference model such as a decision tree. In the construction of the decision tree, all 25 case-level attributes are considered. After being subjected a preprocessing scheme to filter out redundant factors, the remaining attributes are trained to construct a decision tree using the C4.5 algorithm.

3.3 Evaluation Method

For our system of 150 legal cases, we used 10-fold cross validation [32] to validate the model. This method is suitable for small datasets because it applies all instances for both training and validation, and each instance is used for validation exactly once. The final reported accuracies are averaged from all folds.

3.4 Experimental Setup

In accordance with the classifier structure discussed in the previous section, we separated our experiments into two parts: case-level attribute identification and legal-level attribute identification.

Concerning the case-level attribute identification, the experiments were carried out to measure the performance of the proposed model in specifying the class of case-level attributes, as well as to give insights into the performance of the modular ANN compare with that of the conventional ANN. PCA filtering with minimum attribute ranking threshold of 0.2 was applied which has been reported to provide some technical insight. In general, the ANN architecture is set to the back-propagation multilayer neural network with a single hidden layer and the unipolar sigmoid activation function. Depending on the parameter setting of the ANN, the optimal parameters, which are heuristically tuned, are specified be the: (i) number of hidden nodes, (ii) learning rates, (iii) momentums, and (iv) maximum number of epochs.

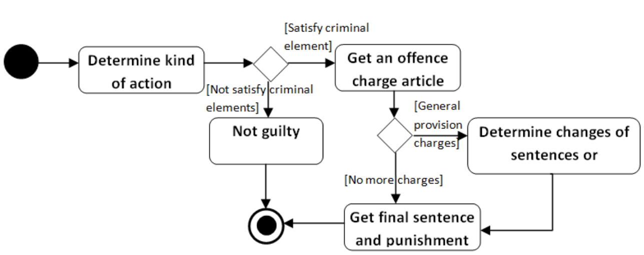


Fig. 6 Charge identification process.

For the legal-level attribute identification, the experiment was designed so that performance of plain C4.5, C4.5 with PCA filtering and C4.5 with SVM ranking are compared. This enables us to gain insights into the 2 input-filtering techniques. The C4.5 decision tree construction method is applied in all experiments with the confidence threshold for tree pruning set at 0.25. Note that, according to our selected law section, the 150 offence cases are separable into 2 groups: 98 harm cases and 52 traffic-related cases. The traffic-related cases are usually complicated owing the incorporation of the traffic law.

4. Results and Discussion

This section presents the results of the experiments and provides insights into the analytical technique and the implications for the law domain.

4.1 Diagnostic Issue Identification with Modular ANN

Tables 6 and 7 show the comparisons of the results obtained from the experiments for the accuracy of our proposed modular ANN and the conventional nonmodular ANN structure. In addition, Table 6 shows the network topology (input-hidden-output) and classification results for all 13 diagnostic issues using the conventional ANN, and Table 7 shows the results for the 6 complex modules that required a modular structure. Note that a number in brackets after a number of

Table 6 Accuracy of the conventional ANN in classifying diagnostic issues.

Net	No. of	Non-PCA		PCA	
	Instances	Topology	Accuracy(%)	Topology	Accuracy(%)
C1	116	22-13-5	77.59%	15-10-5	72.41%
C2	92	27-14-2	79.35%	14-7-2	79.35%
C3	111	20-11-2	95.50%	12-7-2	96.40%
C4	106	23-14-4	83.96%	16-10-4	86.79%
C5	106	28-14-2	73.58%	13-7-2	75.47%
C6	105	18-7-2	95.24%	9-5-2	97.14%
C7	100	5-3-2	83.00%	4-3-2	89.00%
C8	69	3-3-2	78.26%	3-3-2	86.96%
C9	92	13-8-6	76.09%	10-6-6	83.70%
C10	66	4-3-2	80.30%	3-3-2	86.36%
C11	60	6-4-2	95.00%	4-3-2	96.67%
C12	66	6-4-2	96.67%	5-3-2	98.48%
C13	68	5-4-3	79.41%	3-3-3	85.29%

Table 7 Accuracy of the modular ANN in identifying diagnostic issues.

Net	No. of	Non-PCA		PCA	
	Instances	Topology	Accuracy(%)	Topology	Accuracy(%)
C1	116	5(1)-5-5	77.59%	4(1)-5-5	83.62%
C2	92	17(1)-9-2	86.96%	8(1)-5-2	91.30%
C3	111	9(2)-5-2	95.50%	5(2)-3-2	96.40%
C5	106	11(2)-7-2	82.08%	7(2)-5-2	85.85%
C6	105	8(1)-5-2	97.14%	5(1)-3-2	99.05%
C9	92	8(3)-4-6	82.61%	6(3)-3-6	92.39%

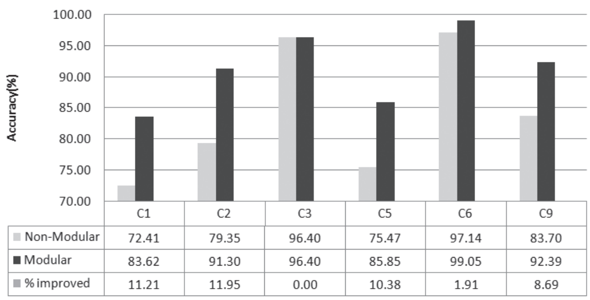


Fig. 7 Improvement in accuracy when PCA is applied to modular ANN.

input nodes refers to the number of inputs from the other modular networks. The network codes are referred to in Table 2. To gain insights into the aspect of input selection for the ANN, the automatic selection of input based on the PCA filtering method is taken into consideration and reported in both tables.

According to Table 6, the accuracy of the regular non-PCA ANN model is in the range of 70-95%. The accuracy is improved by more than 5% on average when the PCA method is applied, except for models C1 (Act_ForceLevel) and C2 (Act_HasChanceToConsider).

In the case of model C1, the accuracy noticeably decreases when PCA filtering is applied (from 77.59% to 72.41%). This suggests that there may be interplay among the attributes and, hence, the application of divide-and-conquer method could be beneficial. Upon the application of modularity, as shown in Table 7 and Fig. 7, significant improvement is obtained in the determining power of case-level issues, raising the overall accuracy to the range of 83.62%-99.05%. Specifically the accuracy of network C9 (Result_Severity) is increased by approximately 10%. This network of three subtasks is a clear example showing that the modular ANN is indeed an effective model.

Interestingly, the accuracy of networks C1 (Act_ForceLevel) and C2 (Act_HasChanceToConsider) is significantly improved when modularity is applied. From the outcomes shown in Table 7, the accuracy of C1 is improved from 77.6% to 83.6%. Similarly, it is improved from 87% to 91.3% for network C2. Also, PCA works well when it is equipped to this divide-and-conquer approach. From the legal viewpoint, all the networks significantly improved by modularity, which are C1 (level of force), C2 (chance to consider), and C5 (planning), involve factors related to the intention of the offender. Intention is one of the most important principles in Thai criminal code (Article 59) and one of the most complicated and subjective legal factors because it is related to the mind of the criminal. This subtle complication was well captured by the process of modular ANN.

Additionally, C3 (cruelty of the action) and C6 (cruelty of concealment) are less complicated and more objective than the 3 above networks. Even the conventional ANN can yield a reasonably good result (over 95%), meaning that it can be only slightly improved by the modular ANN.

Table 8 Accuracy of the C4.5 decision trees with different configurations of data filtering in classifying legal elements.

Model	No. of Instances	Accuracy(%) No Filter	PCA	SVM
ACT-Harm	98	90.82%	87.76%	92.86%
ACT-Traffic	52	98.08%	94.23%	98.08%
ANG	150	95.33%	98.00%	96.00%
AWA	150	98.00%	94.67%	98.00%
CON	150	95.33%	95.33%	96.00%
CRU	150	97.33%	95.33%	97.33%
PLA	150	91.33%	92.67%	91.33%
PRE	150	92.67%	89.33%	95.33%
UNA	150	100.00%	99.33%	100.00%

4.2 Legal Element Identification with Decision Tree

Table 8 shows a comparison of the accuracies of legal element classification models with different configuration of the filtering methods: PCA, SVM, and no filter. In addition to PCA filtering, the SVM ranking algorithm is introduced to enhance the performance of this classifier. The ACT_Intention_Result class is separated into 2 distinct classes, labeled ACT-Harm (action in harm case) and ACT-Traffic (action in traffic case).

The accuracies of C4.5 decision tree model with no filtering are within the range of 90-100%. This means that the standard C4.5 model is capable of identifying the relevant legal elements for this particular set of cases. However, because of the effectiveness of PCA in enhancing the classification performance of case-level attributes, it is interesting to observe the ability of data filtering to enhance the performance of C4.5, especially for complicated classes like ACT class, which classifies the type of action used as the basis for identifying other classes.

When PCA is applied to the models, the accuracy slightly decreases by 1-3% on average compared with the model with no filtering preprocessor. Nonetheless, it can improve the results for 2 classes and has negligible effect on one class. This implies that all the factors input to C4.5 have a similar effect on determining the legal elements. Ineffectively removing any of these factors could adversely affect the overall determining power of the system. In contrast, when applying SVM attribute ranker to support the classifier at the same dimensionality as PCA, the average accuracy slightly improves by up to 3%. The SVM helps to raise the accuracy of the ACT-Harm class from 90.82% to 92.86% and improves the accuracy range to 92.86%-100% for all classes. There are four classes not affected by the SVM ranker. These classes already have 97-100% accuracy giving little margin for improvement. Comparing the SVM to PCA in terms of accuracy enhancement, the SVM is generally superior by up to 6%. SVM is slightly inferior to PCA only in determining the classes of Plan (PLA) and Anger (ANG), by 1.34% and 2.0%, respectively.

To gain an idea of how the decision tree imposition process on this legal reasoning system, let us consider Figs. 8 and 9. Technically, the structure of the deci-

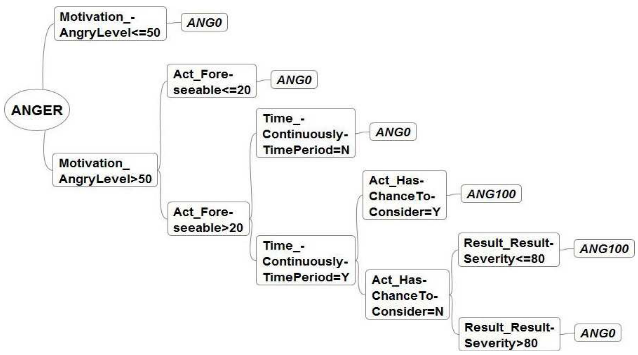


Fig. 8 Example of decision rules specified by C4.5 for "ANGER" class.

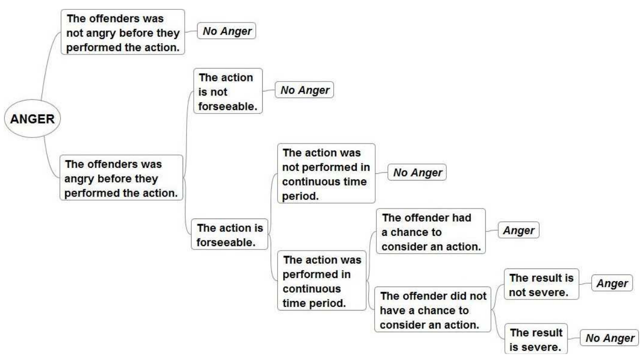


Fig. 9 Interpreted version of decision rules specified by C4.5 for "ANGER" class.

sion tree represents the relationships among objects (tree nodes) based on information entropy theory [6]. In Fig. 8, each node of this sample decision tree is labeled with an enumerated value. This decision tree was constructed by C4.5 with the SVM attribute ranker preprocessor. Referring to the tree in Fig. 9, if the offender were angry before performing the action ($Motivation\_AngryLevel > 50$) and the action was not foreseeable ($Act\_Foreseeable \leq 20$), the offender would be judged as the nonanger condition (encoded as ANG0). Otherwise, if the action was foreseeable ($Act\_Foreseeable > 20$) with a continuous time period ($Time\_ContinuouslyTimePeriod = Y$) and the offender had a chance to consider the action ($Act\_HasChanceToConsider = Y$), the offender would be judged as having an angry state of mind (encoded as ANG100).

The interpreted version of this tree is shown in Fig. 9. This tree shows the possible paths of factor determination to obtain the legal-level attribute and also represents the conceptual knowledge obtained from the model.

4.3 System Integration and Technical Insights

To gain more insights into the accuracy of the overall system, the two-stage classifiers, the complete system comprising modular ANN with PCA filtering and the C4.5 decision tree with the SVM attribute ranker, were executed thoroughly. By putting the intermediate result of case-level attributes transparent, the accuracy of identifying each legal element was directly measured against the provision of case fact input. The obtained results are shown in Table 9.

Table 9 Accuracy of the integrated system with the proposed two-stage classifier.

Legal-level attributes	Selected case-level attributes (mANN)	Accuracy (%)	Change (%)
ACT-Harm	C1, C2, C5, C6, C7, C8, C9	90.82	-2.04
ACT-Traffic	C9, C11, C12	96.15	-1.92
ANG	C2, C9, C13	94.67	-1.33
AWA	C4, C5	98	0.00
CON	C6, C9, C10, C13	94	-2.00
CRU	C1, C9	97.33	0.00
PLA	C5, C6, C13	90.67	-0.67
PRE	C1, C3, C4, C5, C9, C13	91.33	-4.00
UNA	C7, C10	98	-2.00

Table 10 Accuracy of the integrated system driven by the hold-out validation method.

Model	Cases	Input dimensions	Output dimensions	Accuracy (%)
ACT-Harm	18	118	7	14 (77.78%)
ACT-Traffic	7	28	3	6 (85.71%)
ANG	25	48	2	20 (80.00%)
AWA	25	51	3	24 (96.00%)
CON	25	43	2	22 (88.00%)
CRU	25	37	2	24 (96.00%)
PLA	25	51	2	21 (84.00%)
PRE	25	113	6	19 (76.00%)
UNA	25	9	2	22 (88.00%)

According to Table 9, the accuracy of the integrated system is satisfactory with an average accuracy of 94%, although this is slightly lower than that of the independent classifiers. This is expected to be due to the chaineffect caused by a false decision in an early stage of case-level attribute classification.

Furthermore, in addition to the 10-fold cross-validation method, a hold-out method [32] is also applied to evaluate the integrated model with 25 additional unseen cases to investigate the efficiency of the obtained model. These new cases were selected from verdicts appearing in 1989. The results of the evaluation are shown in Table 10. The overall accuracy is satisfactory (84% on average). Note that the models of ACT-Harm and PRE have the lowest accuracies of 78% and 76%, respectively. These results are consistent with those obtained from the 10-fold test scheme. The lower accuracy compared with that obtained in the validation stage, exposes a minor issue of undertraining. This is expeced for the cases of limited training instances. ACT-Harm and PRE models exhibited the largest drops in accuracies since they require much higher dimensionality of the input and output than the other models.

4.4 Application of System in Legal Domain

To illustrate the application of the proposed model, we selected a case (verdict no. 1729/2547) which is summarized in Fig. 10. The factor hierarchy shown in Fig. 11 was obtained by classification using the Result_Severity (C9) network with the selected test set. In this case, all the diagnostic issues are correctly classified and labeled as the output of each module. This figure only illustrates the outputs of the modules important for describing the case. The resulting analysis can be interpreted as: "The damaged organ was important, the weapon used was dangerous, and the injury was severe. When these factors are considered with other independent factors, the severity of the result is considered to be physically harmful (*PhysicalSevere*)".

According to the results of the proposed method, the resulting diagnostic issues were next subjected to legal element identification. To illustrate this, we selected the most complex tree, ACT-Harm (action in harm case) to produce a path that reflects the decision rules in the judicial process. The complete decision tree for this class and a summary of the path trraversing the tree in this case are shown in Figs. 12 and 13, respectively. Finally, all of the collected

-The defendant used a knife with a handle 16.5 cm long, 48 cm long and 31.5 cm wide blade 6 cm as a weapon to rapidly pierce and slash the victim, Mr. Akom, near the shoulder and the back of the head, causing the injury.
-The defendant's 73 year-old grandfather was also injured by the knife from the defendant while he was attempting to harm the victim again.
-As a result of the intervention, the defendant fled.
-A doctor commented that the victim will recover in 2 weeks in the case of no complications.

Fig. 10 Summary of the case leading to verdict no. 1729/2547.

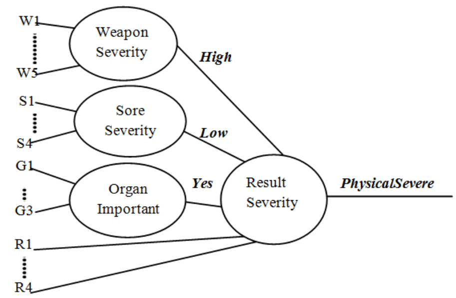


Fig. 11 Derived diagnostic issues for verdict no. 1729/2547.

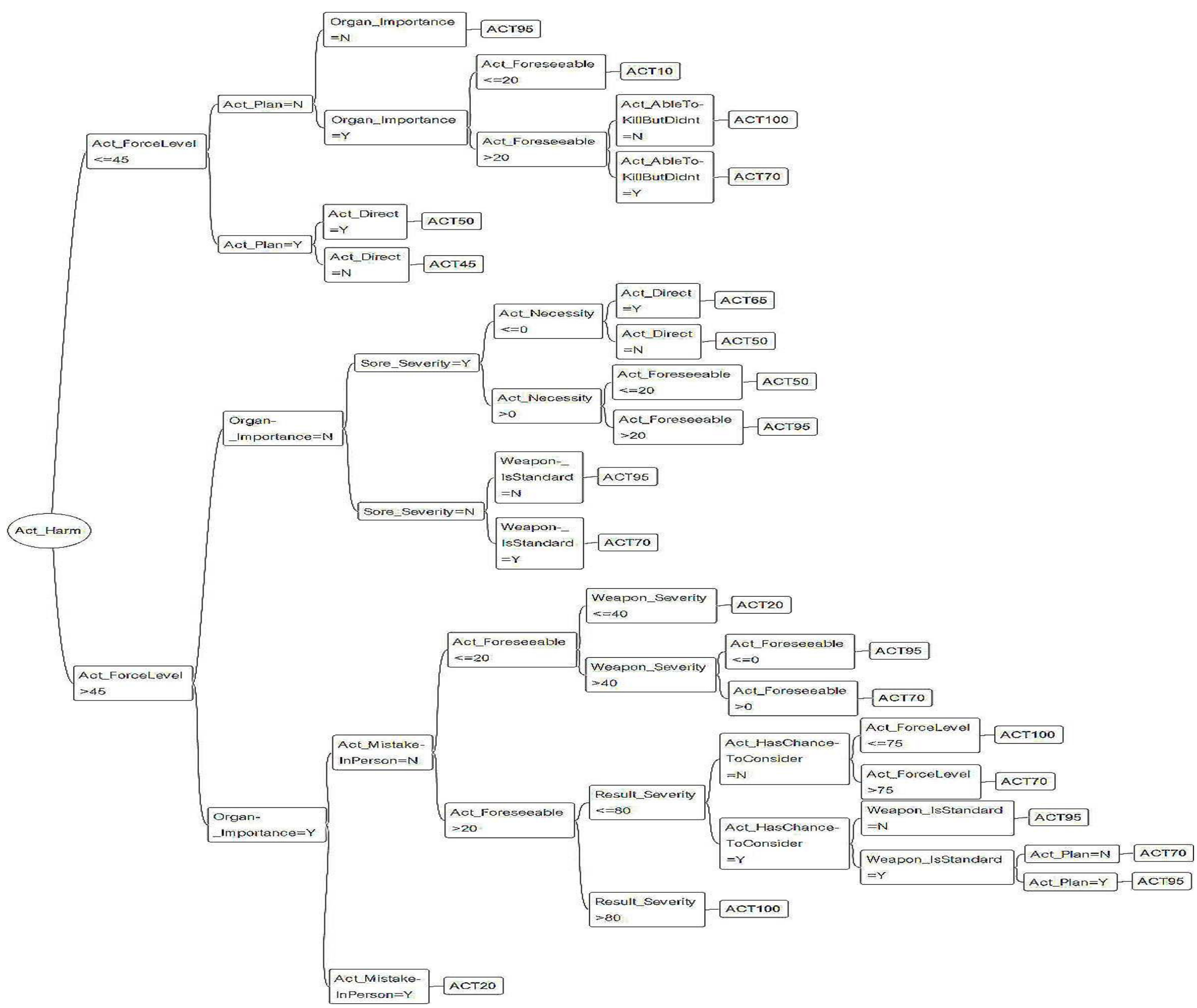


Fig. 12 Final decision tree for Act-Harm class.

```
-Step 1) It was not possible for the action to kill
the victim. (Act_ForceLevel ≤ 45)
-Step 2) The action was performed with planning.
(Act_Plan = Y)
-Step 3) The action was directly performed to the
victim. (Act_Direct = Y)
-Result: The action was intended harm but had no
severe result. (ACT50)
```

Fig. 13 Interpretation of ACT-Harm class for verdict no. 1729/2547.

legal elements are mapped to the legal charges and punishment ranges. In this case, the final charge is Article 295, which has a sentence of 2 years of imprisonment. The actual result of this case was that the defendant was imprisoned for 2 years. Readers can download the decision trees for all models at http://zotarat.com/tree.htm.

5. Conclusions and Future Works

We proposed a system that uses a two-stage classifier to identify legal charges and the punishment range given factual case data. The first stage abstracts the facts from the criminal case. Subsequently, the second stage identifies legal elements associated with the legal charges and punishment range.

Technically, the models of the modular ANN with PCA preprocessor, and the C4.5 decision tree with the SVM-ranker model satisfactorily deliver the functions of identifying diagnostic issues and relevant legal elements, respectively. The first classifier of the diagnostic issue identification comprises seven independent ANN models that feed the output to the other 6 ANN models in modular organization. Using the obtained diagnostic issues, the second classifier uses the C4.5 decision model to identify the target legal element from among the 8 candidates.

A total of 150 criminal cases, limited to only the offences against the life and body section under Thai criminal law codes were selected into TCXML data management system for experiment. To cope with such a limited number of cases for the learning process, a set of factor analysis techniques, such as PCA and attribute ranker using an SVM, were applied to alleviate the undertraining issue and also help to squeeze the final results.

In the experiments, the result of 84% accuracy obtained after the first stage proves that the modular ANN is an effective model for classifying the diagnostic issues. By applying appropriate preprocessing methods such as PCA, the performance was further improved to 92% accuracy. The accuracy 95% obtained after the second stage also shows that the C4.5 decision tree is appropriate for classifying the legal elements based on the preidentified case-level attributes obtained in stage one. The experiments also show that the SVM attribute ranker is more appropriate for preprocessing the input data than PCA. More importantly, our integration test of the multistage classifier clearly proved that the system is effective for determining legal elements that identify the corresponding charges and ranges of penalty. As illustrated in a sample case, the traceability of C4.5 and the modular ANN structure enable the causes and effects for the identified charges and penalty range to be clearly interpreted.

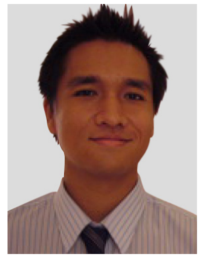
However, there are some challenging issues for future works. In this study, some theoretical principles are suppressed, e.g. innocent agent, principals and supporters, mistake, etc. Extending the study to cover these principles requires involvement of law experts and extensive data modeling. Moreover, since the scope of this classification system is limited to the offences against life and body section of the criminal law code only, extending this classification model to support others or all criminal law articles is a great challenge. Finally, the final results should be extended to a specific amount of punishment instead of set of charges which provided only the theoretical punishment amounts.

In response to such the forthcoming domain complexity and huge input dimensionality, the comprehension of a more sophisticated method and the engagement of the law experts, to reliably validate the assumptions and organization of the model, is also a challenge to come.

References

[1] C. Pejovic, "Civil law and common law: two different paths leading to the same goal," Victoria University Wellington Law Review, pp.817–841, 2001.
[2] S. Thammaboosadee, B. Watanapa, and N. Charoenkitkarn, "A framework of multi-stage classifier for identifying criminal law sentences," Procedia Computer Science, vol.13, pp.53–59, 2012.
[3] K.D. Ashley, "An AI model of case-based legal argument from a jurisprudential viewpoint," J. Artificial Intelligence and Law, vol.10, pp.163–218, 2002.
[4] A. Schmidt and Z. Bandar, "Modularity: a concept for new neural network architectures," Proc. IASTED International Conf. Computer Systems and Applications, pp.26–29, Jordan, 1998.
[5] J.R. Quinlan, C4.5: Programs for machine learning, Morgan Kaufmann Publishers, San Francisco, 1993.
[6] S. Thammaboosadee and B. Watanapa, "Identification of criminal case diagnostic issues: a modular ANN approach," Int. Journal of Information Technology and Decision Making, vol.12, no.3, pp.523–546, 2013.
[7] S. Kullback, Information theory and statistics, John Wiley & Sons, New York, 1959.
[8] H. Deng, G. Runger, and E. Tuv, "Bias of importance measures for multi-valued attributes and solutions," Proc. 21st International Conf. Artificial Neural Networks, pp.293–300, 2011.
[9] J. Gutierrez and G. Leroy, "Using decision trees to predict crime reporting," in Advanced Principles for Improving Database Design, Systems Modeling, and Software Development, eds, J. Erickson and K. Siau, IGI Global, pp.132–145, 2008.
[10] M. Craven and J. Shavlik, "Using neural networks for data mining," Future Generation Computer Systems, vol.13, pp.211–229, 1997.
[11] G.C. Oatley, B.W. Ewart, and J. Zeleznikow, "Decision support systems for police: lessons from the application of data mining techniques to 'soft' forensic evidence," Proc. 24th SGAI International Conf. on Knowledge Based Systems and Applications of Artificial Intelligence, pp.35–100, 2006.
[12] M.M.J. Theresa and V.J. Raj, "Analogy making in criminal law with neural network," Proc. International Conf. Emerging Trends in Electrical and Computer Technology, pp.772–775, 2011.
[13] E. Aleskerov, B. Freisleben, and B. Rao, "CARDWATCH: a neural network based database mining system for credit card fraud detection," Computational Intelligence for Financial Engineering,

pp.220–226, 1997.
[14] K. Swingler, Applying neural networks: a practical guide, London Academic Press, 1996.
[15] L. Yi and H.R. Thomas, "A decision support system for the environmental impact of ICT and e-business," Int. J. Information Technology & Decision Making, vol.2, pp.361–367, 2009.
[16] S.C. Watanapa, B. Thipakorn, and N. Charoenkitkarn, "A sieving ANN for emotion-based movie clip classification," IEICE Trans. Inf. Syst., vol.E91-D, no.5, pp.1562–1572, May 2008.
[17] H. Zhu, I. Yoshihara, K. Yamamori, and M. Yasunaga, "Multimodal neural networks for symbolic sequence pattern classification," IEICE Trans. Inf. & Syst., vol.E87-D, no.7, pp.1943–1952, July 2004.
[18] A. Stranieri and J. Zeleznikow, "Knowledge discovery from legal databases," Law and Philosophy Library, vol.69, 2005.
[19] J. Zeleznikow, "Using web-based legal decision support systems to improve access to justice," Information & Communications Technology Law, vol.11, no.1, pp.15–33, 2002.
[20] I.H. Witten and E. Frank, Data mining: practical machine learning tools and techniques, 2nd ed., Morgan Kaufmann, 2005.
[21] S. Thammaboosadee and U. Silparcha, "TCXML for collection of verdict of Thai Dika court," (in Thai), Proc. National Conf. Information Technology, pp.179–186, Thailand, 2006.
[22] T. Bray, J. Paoli, C.M. Sperberg-McQueen, and E. Maler, "Extensible markup language (XML) 1.0," W3C Recommendation, http://www.w3.org/TR/REC-xml, accessed Oct. 2010.
[23] United States Department of Justice, "Global Justice XML Data Model," http://it.ojp.gov/jxdm., accessed Jan. 2011.
[24] N. Ye, "The handbook of data mining," Mahwah, Lawrence Erlbaum, New Jersey, 2003.
[25] I.T. Jolliffe, "Principal component analysis," Springer Series in Statistics, 2nd edition, Springer, NewYork, 2002.
[26] V.N. Vapnik, Statistical learning theory, Wiley Interscience, 1998.
[27] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," Machine Learning, pp.389–422, 2002.
[28] F. Li and Y. Yang, "Analysis of recursive feature elimination methods," Proc. 28th annual Inernational ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '05), pp.633–634, 2005.
[29] L.V. Maaten, E. Postma, and J.V. Herik, "Dimensionality reduction: A comparative review," Technical Report TiCC-TR 2009-005, Tilburg University, the Netherlands, 2009.
[30] C. Asaro, M.A. Biasiotti, P. Guidotti, M. Papini, M.T. Sagri, and D.A. Tiscornia, "Domain ontology: Italian crime ontology," Workshop on Legal Ontologies, pp.1–7, 2003.
[31] Y. Yuthankun, The criminal code: translated Thai-English, Soutpaisal Publisher, Bangkok, 2009.
[32] P.A. Devijver and J. Kittler, Pattern recognition: A statistical approach, Prentice Hall, London, 1982.



Sotarat Thammaboosadee was born in 1982 and received his B.Eng. and M.Sc. degrees in Computer Engineering and Technology of Information System Management from Mahidol University, Thailand, in 2003 and 2005, respectively. He also received a Ph.D. in Information Technology from King Mongkut's University of Technology Thonburi (KMUTT), Thailand in 2013. He is now a lecturer at Technology Information System Management Division at the Faculty of Engineering, Mahidol University, Thailand. His research interests include data mining in several domains such as the legal domain, transportation domain, and healthcare domain.

Bunthit Watanapa received his B.Eng. in Computer Engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand, in 1987. He then received his M.Eng. and Ph.D. in Industrial Engineering from Asian Institute of Technology (AIT), Thailand, in 1990 and 2003, respectively. He is now a lecturer and the chairperson of the Business Information System program at the School of Information Technology, King Mongkut's University of Technology Thonburi (KMUTT), Bangkok, Thailand.

Jonathan H. Chan is the Associate Dean for International Relations and the Chairperson of the BSc Computer Science program at the School of Information Technology, King Mongkut's University of Technology Thonburi (KMUTT), Thailand. Jonathan holds a B.A.Sc., M.A.Sc., and Ph.D. degree from the University of Toronto. He is a senior member of IEEE, a member of the IEEE CIS Emergent Technologies Technical Committee (ETTC), and a member of Professional Engineers of Ontario (PEO). His research interests include mathematical modeling and simulation, bioinformatics, parallel computing, and machine learning.

Udom Silparcha is one of the pioneer researchers in Legal Informatics and Computational Law in Thailand. He received a B.Eng. in Computer Engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand in 1987, an M.Sc. in Computer Science from Ball State University, IN, USA in 1991, and a Ph.D. in Computer Science from University of New South Wales, Australia in 1999. He then got an LL.B. from Thammasat University, Thailand in 2004. He is now a lecturer and vice chairperson of Computer Science program at School of Information Technology, King Mongkut's University of Technology Thonburi, Bangkok, Thailand.