A GUI Prototype for the Framework of Criminal Judicial Reasoning System[*]

Sotarat Thammaboosadee[1] and Udom Silparcha[1]
[1]School of Information Technology
King Mongkut's University of Technology
Thonburi, Bangkok, Thailand
(sotarat, udom)@sit.kmutt.ac.th

Abstract: This paper proposed a developed graphical user interface (GUI) prototype, which is supported by the framework of data mining techniques-based criminal judicial reasoning system. The GUI sequences of the prototype are satisfied with criminal judicial procedure in civil law system. Initially, user must build the model by input the existing incident and specifying the detail of objects, elements of crime, charge and judgment. After enough training, the prototype will be ready to determine judgments from new occurred incidents. The prototype shows only the results of each module which help in the decision process. This GUI prototype is useful with lawyers, courts or other people who want to determine the guilt, charges and judgments in their incidents.

1. Introduction

According to the civil law system, the judicial proceedings are strictly based on an abstract rule for applying to the cases. This is different from common law system since the common law draws abstract rules from cases[1]. The difference between civil law and common law lies not just in codification, but in the methodological approach to codes and statutes. The court's judgments are based on the provisions of codes and statutes, from which solutions in particular cases are to be derived. Courts thus have to reason extensively on the basis of general rules and principles of the code. By contrast, in the common law system, cases are the primary source of law, while statutes are only seen as incursions into the common law and thus interpreted narrowly.

In our previous findings, we had proposed a framework of criminal judicial reasoning system using data mining techniques[2]. This framework proposed a data mining methodology based system that focus on criminal cases in civil law system. This framework consists of three main modules which are supported by the judicial procedure of the civil law. A set of incidents in Thai Court XML (TCXML) [3] format is an input of the system and the final output is a set of sentence for each defendants.

In this paper, we introduce a graphical user interface (GUI) prototype for the criminal judicial reasoning system. The prototype is based on two major approaches. The first approach is the satisfaction by our previous proposed framework and the second is the data mining processes, including training and testing processes. The remainder of this paper is organized as follow: The next section shows the related works that provide ideas or model of the user interface in legal domain. Then, section 3 introduces our proposed framework that this prototype is based on, the criminal judicial reasoning system. Then the next section proposed the model of the prototype. In section 5, the result and discussion will be shown and final section 6 contains the conclusion.

2. Related Works

The concept of the intelligence legal interfaces was introduced by F. Borges, D. Bourcier, E. Andreewsky and R.Borges (2001) in "Conception of cognitive Interfaces for legal knowledge: Evolution of the JURISQUE project on the risks of avalanches" [4]. They introduced model-based legal information retrieval interface. The purpose of

[*] This article was originally published in Kierkegaard, S. (2008) Synergies and Conflicts in Cyberlaw. IAITL .pp.292-299

[1] Apple, J.G. and Deyling, R.P. (1995) A Primer on the Civil-Law System, Washington, D.C., Federal Judicial Center.

[2] Thammaboosadee, S. and Silparcha, U. (2008), A Framework for Criminal Judicial Reasoning System using Data Mining Techniques, The 2008 Second IEEE International Conference on Digital Ecosystems and Technologies (IEEE-DEST 2008), Phitsanulok, Thailand, pp.700-705.

[3] Thammaboosadee, S. and Silparcha, U. (2006) TCXML for Collection of Verdicts of Thai Dika Court (in Thai), The National Conference on Information Technology 2006 (NCIT2006), Bangkok, Thailand, pp.179-186.

[4] Borges, F., Bourcier, D., Andreewsky, E.and Borges, R. (2001) Conception of cognitive Interfaces for legal knowledge – Evolution of the JURISQUE project on the risks of avalanches, ICAIL 2001 The Eighth International Conference on Artificial Intelligence and Law.

this model is to connect and evaluate the existing online legal database, which has different approaches in management of various resources and their communication. This interface is called cognitive because it takes into account a meta-knowledge implied in the situation of dialogue with a self-adaptation to types of specific requests in a domain.

There is one extension of the above model. M. D.Rosnay[5] proposed that the cognitive interface applies to copyrighted work online sharing and transaction. The application attention will be dedicated to users' expectations in terms of problem description and legal situations expression. In order to be interoperable and fair, Digital Rights Management systems need a common vocabulary describing legal use cases.

3. Criminal Judicial Reasoning System using Data Mining Techniques

Our paper, "Criminal Judicial Reasoning System using Data mining techniques" proposed a framework in a data mining-based judicial system. This framework consists of three main modules based on civil law system. An overall framework model is shown in figure1. The input of the system is a sequence of occurred incident which are structured in TCXML (Thai Court XML). The TCXML is an adaptation of Global Justice XML Data Model [6] which has a purpose in Thai verdict document collection. All of verdict information, also details of an incident, are collected in TCXML is semantic structure. Then each input incident will be classified for the elements of crime [7] by the "Elements of Crime Parser". A classification data mining algorithm is proper for this module. In this framework, the decision tree algorithm was selected. This decision tree algorithm is based on XML dataset. Therefore, each attribute of this tree has to be classified by XML tree similarity determining. After identifying each attribute value, the model of the elements of crime will be trained. The elements of crime can be mainly categorized into three elements: act, external elements and internal element. These three main elements also can be categorized, such as type of acts, type of person, relationship of the person, used property and weapon, type of motivation, etc. Each classified elements of crime will be mapped for the law code by "Charge Mapper". This module is driven by a table-looking up method. Then the "Sentence Discoverer" will find the sentence for each defendant. The kind of sentences and their severity will be considered using charges and the details of the incidents. The factors that were not identified in Elements of Crime will be re-identified from a TCXML set. This module is driven by the fuzzy regression method that has two main input sources: additional incident factors from an TCXML incident and the type and range of punishment for each charge code. A detail of input for each module is described in table1.

Table1: The details of input and output of the system.

Name	Input	Output
Incident in TCXML format.	Elements of Crime Parser	
Sets of Elements of Crime Value of each incident.	Charges Mapper	Elements of Crime Parser
Charge No. Of each defendant and original incident TCXML.	Sentences Discoverer	Charges Mapper
Punishment type and amount of each defendant.		Sentences Discoverer

[5] Rosnay, M.D. (2003) Cognitive Interfaces for Legal Expressions Description – Application to Copyrighted Works Online Sharing and Transactions in D. Bourcier (ed.), Legal Knowledge and Information Systems. Jurix 2003: The Sixteenth Annual Conference. Amsterdam: IOS Press, 2003, pp. 121-130.

[6] Justice XML Structure Task Force, The Justice XML Data Model: Overview and Status, http://it.ojp.gov/jxdm.

[7] Vachanasvasti, K. (2006) Criminal Law: Part 1 (in Thai), Thammasat university Publishing.

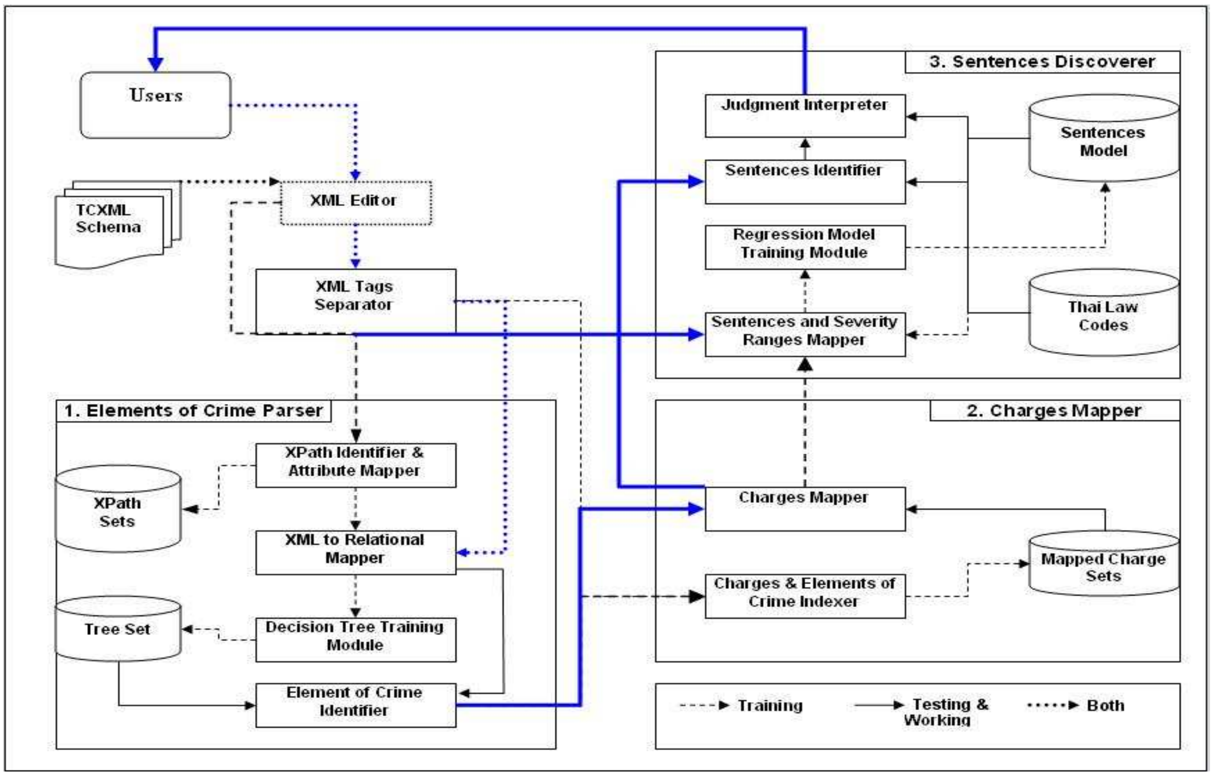


Figure1. An overall process in the frame work of the criminal judicial reasoning system.

The scope of the framework considered only the occurred incidents. The other factors which do not involve the input incidents, such as proof of witness or mitigation, are not included in this framework.

4. The Prototype Model

This prototype is developed as a windows application in Dot Net 2.0 framework. The model of this prototype is based on figure1. This prototype has two main modules, which are model builder and testing modules, while the additional module is the law code viewer. Both model builder and testing modules have three common sub-modules. The difference is the results of each sub-module in model builder will be justified by users. These adjusted results are used to build a model. The first module is Elements of Crime Parser. The input incident can be edited via the extension of XMLGridControl component which is an open source schema-based xml editor control. The XMLGridControl is developed by Michael Coyle [8][8]. The input TCXML incident will be extracted for the incident and their information such as offender, victim, type of act, etc. After this extraction, there are two tasks to categorized elements of crime. The first task is the extraction of sub-elements for each incident such as type of acts, type of defendants or type of intentions. Each sub-element may have one or more values. Another task is the categories of each object, which are persons, organizations, properties and locations. These categories are depended on the criminal law part 2[9]. After adjustment, the model of elements of crime will be trained. The second main module is Charge Mapper. User can set the charge codes to the defendants. Each law code number can be viewed according to its description. After charge code setting and then training, user can set the sentences and their amounts for each defendant. Another part of the prototype is model tester module. The interfaces of this module are quite similar with model builder except the results are not adjustable.

5. Results and Discussion

The user interfaces of the prototype are shown in figure 2-8. The main page of the application is shown in figure 3. Users can select three choices from this page. The last one in Charge Code Viewer that user can view and search the details of each criminal law code.

[8] Coyle, M., XmlGridControl Editing XML Data in the PropertyGrid, from http://www.codeproject.com/KB/grid/XmlGridControl.aspx

[9] Likasitwattanakul, S. (2008) Criminal Law Code (in Thai), Winyuchon Publishing.

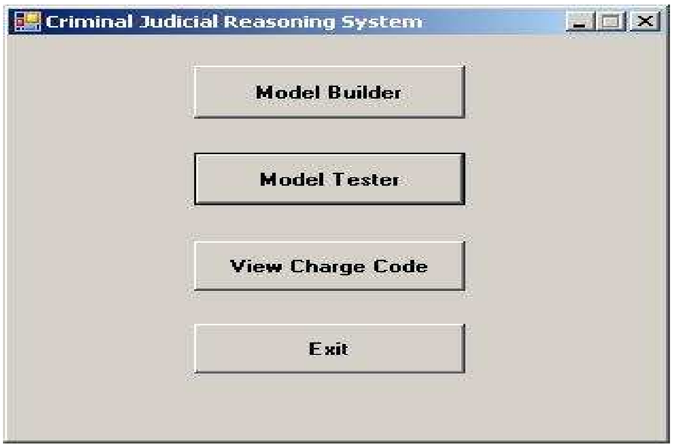
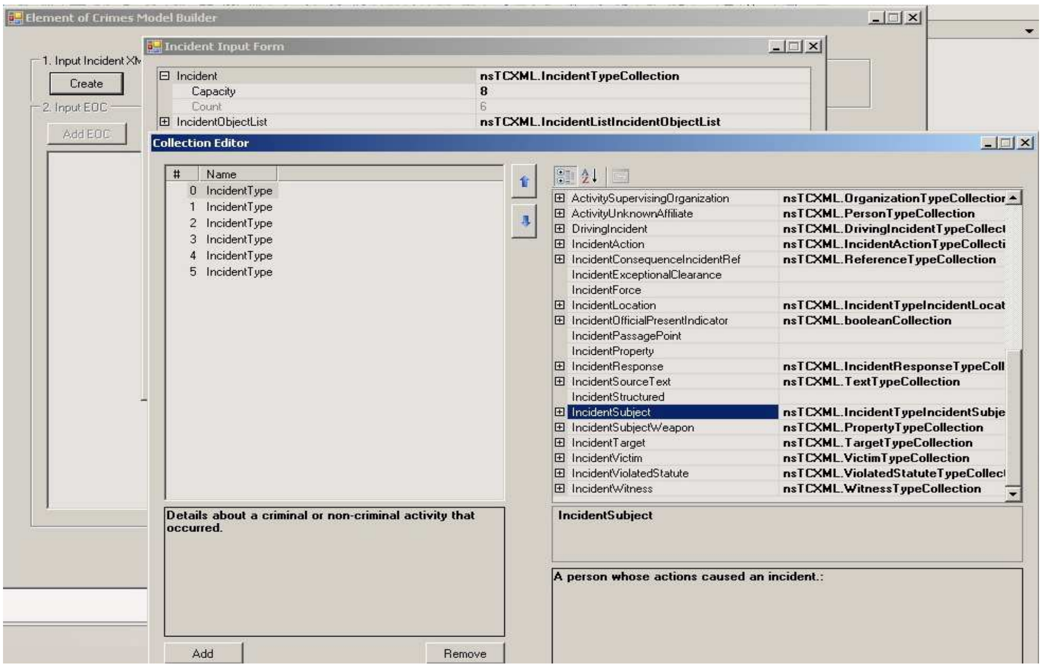


Figure2. The main page of the prototype

In Model Builder module, the first task is the Element of Crime Training module. User must create or load a set of occurred incidents which is in TCXML format. In this demonstration, we will use the incident in verdict No. 234/2538. The interface of incident input is shown in figure3 below.



⊟ Incident	nsTCXML.IncidentTypeCollection
Capacity	8
Count	6
⊞ IncidentObjectList	nsTCXML.IncidentListIncidentObjectList

#	Name
0	IncidentType
1	IncidentType
2	IncidentType
3	IncidentType
4	IncidentType
5	IncidentType

⊞ ActivitySupervisingOrganization	nsTCXML.OrganizationTypeCollectior
⊞ ActivityUnknownAffiliate	nsTCXML.PersonTypeCollection
⊞ DrivingIncident	nsTCXML.DrivingIncidentTypeCollect
⊞ IncidentAction	nsTCXML.IncidentActionTypeCollecti
⊞ IncidentConsequenceIncidentRef	nsTCXML.ReferenceTypeCollection
IncidentExceptionalClearance	
IncidentForce	
⊞ IncidentLocation	nsTCXML.IncidentTypeIncidentLocat
⊞ IncidentOfficialPresentIndicator	nsTCXML.booleanCollection
IncidentPassagePoint	
IncidentProperty	
⊞ IncidentResponse	nsTCXML.IncidentResponseTypeColl
⊞ IncidentSourceText	nsTCXML.TextTypeCollection
IncidentStructured	
⊞ IncidentSubject	nsTCXML.IncidentTypeIncidentSubje
⊞ IncidentSubjectWeapon	nsTCXML.PropertyTypeCollection
⊞ IncidentTarget	nsTCXML.TargetTypeCollection
⊞ IncidentVictim	nsTCXML.VictimTypeCollection
⊞ IncidentViolatedStatute	nsTCXML.ViolatedStatuteTypeCollect
⊞ IncidentWitness	nsTCXML.WitnessTypeCollection

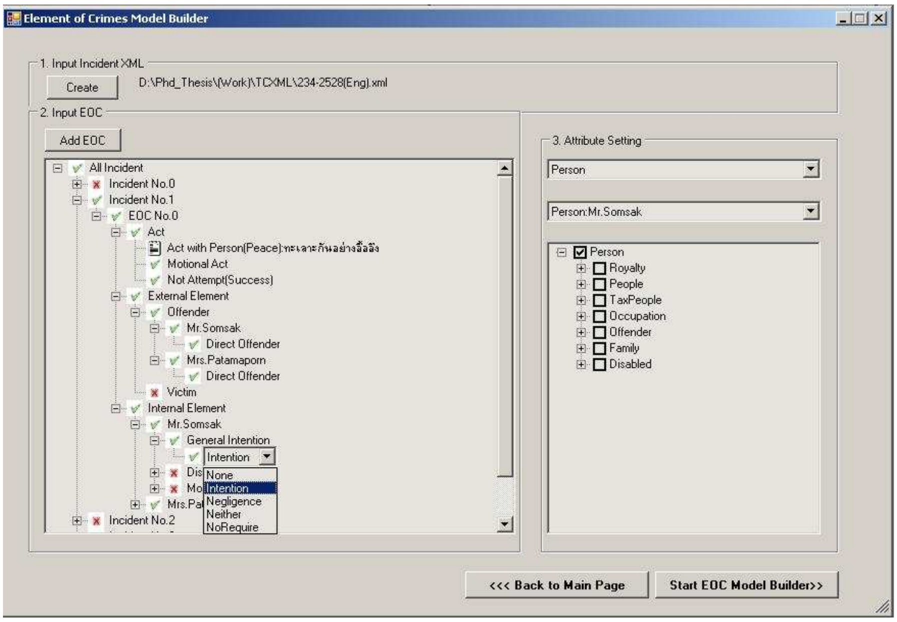
Details about a criminal or non-criminal activity that occurred.

IncidentSubject

A person whose actions caused an incident.:

Figure3. The incident TCXML editor.

After all incidents are entered, the information of each incident will be extracted for involved objects. The objects are composed of persons, organizations, properties and locations. The element of crime of each incident can be adjusted in the dropdown tree view control on the left panel and the category of each extracted object can be selected and set at the right panel of this page. As shown in figure5 below.



D:\Phd_Thesis\(Work)\TCXML\234-2528(Eng).xml

```
⊟ ✓ All Incident
   ⊞ ✗ Incident No.0
   ⊟ ✓ Incident No.1
      ⊟ ✓ EOC No.0
         ⊟ ✓ Act
               Act with Person(Peace):ทะเลาะกันอย่างอื้ออึง
            ✓ Motional Act
            ✓ Not Attempt(Success)
         ⊟ ✓ External Element
            ⊟ ✓ Offender
               ⊟ ✓ Mr.Somsak
                     ✓ Direct Offender
               ⊟ ✓ Mrs.Patamaporn
                     ✓ Direct Offender
               ✗ Victim
         ⊟ ✓ Internal Element
            ⊟ ✓ Mr.Somsak
               ⊟ ✓ General Intention
                     ✓ Intention ▼
               ⊞ ✗ Dis
               ⊞ ✗ Mo
            ⊞ ✓ Mrs.Pa
   ⊞ ✗ Incident No.2
```

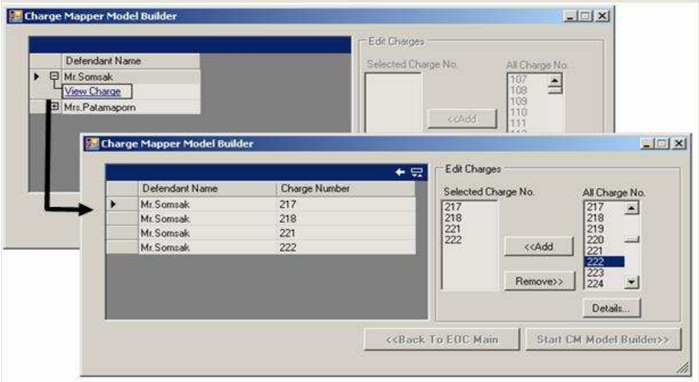
None
Intention
Negligence
Neither
NoRequire

Person
Person:Mr.Somsak

```
⊟ ☑ Person
   ⊞ □ Royalty
   ⊞ □ People
   ⊞ □ TaxPeople
   ⊞ □ Occupation
   ⊞ □ Offender
   ⊞ □ Family
   ⊞ □ Disabled
```

Figure4. The Elements of Crime Training module.

After adjustments, the elements of crime model will be trained and go through the Charge Mapper module. According to figure5, user can set the charge code for each defendant and also can view the details of each law code. The charge model will be trained when the law code adjustment is finished. Then user can set the sentences and amounts for each defendant in Sentence Discoverer Model Builder, which is the final module in the model builder, as shown in figure 6.



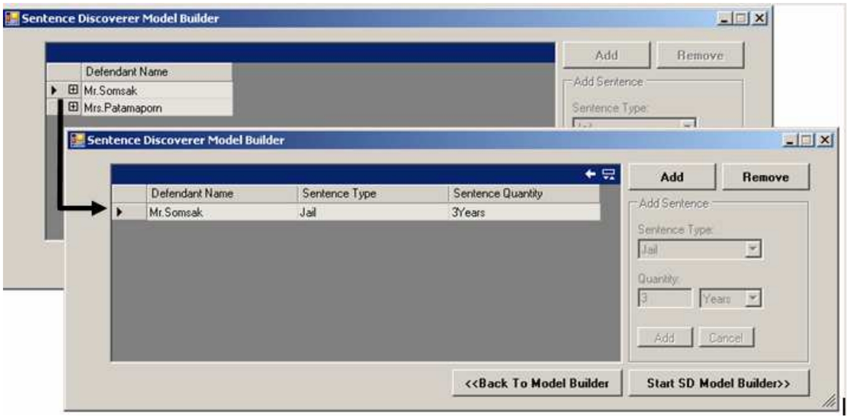
Defendant Name
⊟ Mr.Somsak
View Charge
⊞ Mrs.Patamaporn

Defendant Name	Charge Number
Mr.Somsak	217
Mr.Somsak	218
Mr.Somsak	221
Mr.Somsak	222

Selected Charge No.: 217, 218, 221, 222

All Charge No.: 217, 218, 219, 220, 221, 222, 223, 224

Figure5. Charge Mapper Training Module.



Defendant Name
⊞ Mr.Somsak
⊞ Mrs.Patamaporn

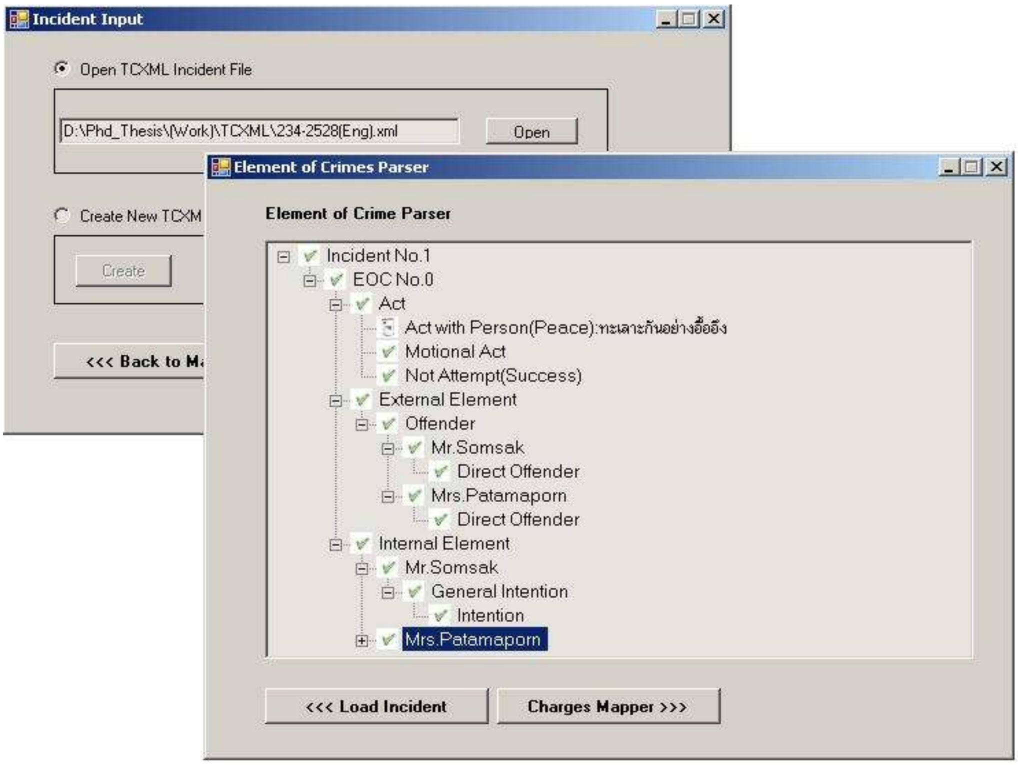
Defendant Name	Sentence Type	Sentence Quantity
Mr.Somsak	Jail	3Years

Sentence Type: Jail

Quantity: 3 Years

Figure6. Sentence Discoverer Training Module

In Model Testing module, trained models from Model Builder will be used to discover the result in each module. We will use a same incident set as model builder in the following demonstration. At first, user must enter the incidents which can be newly created or he can open an existing document. Then the Elements of Crime Parser will show the elements of crime value in treeview control and show also their description text, as shown in figure7.

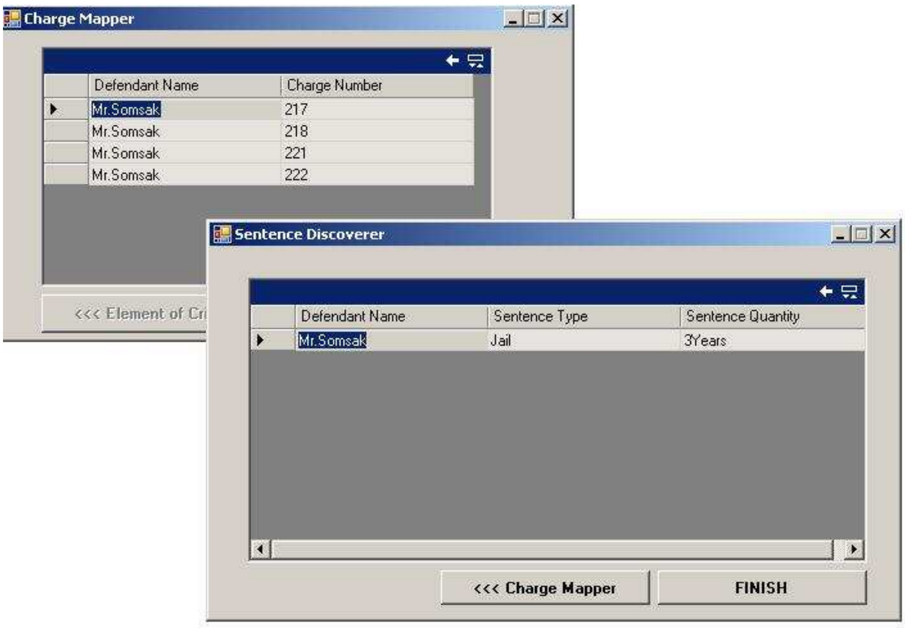


D:\Phd_Thesis\(Work)\TCXML\234-2528(Eng).xml

- ✓ Incident No.1
  - ✓ EOC No.0
    - ✓ Act
      - Act with Person(Peace):ทะเลาะกันอย่างอื้ออึง
      - ✓ Motional Act
      - ✓ Not Attempt(Success)
    - ✓ External Element
      - ✓ Offender
        - ✓ Mr.Somsak
          - ✓ Direct Offender
        - ✓ Mrs.Patamaporn
          - ✓ Direct Offender
    - ✓ Internal Element
      - ✓ Mr.Somsak
        - ✓ General Intention
          - ✓ Intention
      - ✓ Mrs.Patamaporn

Figure7. Elements of Crime Parser in Testing Module.

Finally, the Charge Mapper and Sentence Discoverer have a similar interface with Model Builders' except their results are not adjustable. The list of charge numbers, sentences types and their amounts for each defendant will be shown in these parts, as shown in figure8.



Defendant Name	Charge Number
Mr.Somsak	217
Mr.Somsak	218
Mr.Somsak	221
Mr.Somsak	222

Defendant Name	Sentence Type	Sentence Quantity
Mr.Somsak	Jail	3Years

Figure8. Charge Mappper and Sentence Discoverer in Testing Module.

6. Conclusion

This paper proposed a developed GUI prototype, which is supported by the framework of criminal judicial reasoning system. The processes of the framework satisfy the criminal judicial procedure in civil law system. The prototype, which has an approach to support this framework, also has the same main processes and sequences. To satisfy the data mining processes, these three processes are separated into two modules - Model Builder module and Model Testing module. User must build the model by inputting the existing incident and specifying the detail of objects, elements of crime, charges and judgments. After the model had been sufficiently trained enough, the Testing module is ready to determine the sentences from the incidents set. This GUI prototype can be useful with lawyers, court or other people who want to determine the judgment upon their occurred incidents. This prototype shows only the results of each module. The detail of the reasoning process, which is the source of the results, is not shown to user. This may lead to a future work in developing an application that developed for interactive GUI for representing the details in reasoning process.

References

1) Apple, J.G. and Deyling, R.P. (1995), A Primer on the Civil-Law System, Washington, D.C., Federal Judicial Center.
2) Thammaboosadee, S. and Silparcha, U. (2008), A Framework for Criminal Judicial Reasoning System using Data Mining Techniques, The 2008 Second IEEE International Conference on Digital Ecosystems and Technologies (IEEE-DEST 2008), Phitsanulok, Thailand, pp.700-705.
3) Thammaboosadee, S. and Silparcha, U. (2006), TCXML for Collection of Verdicts of Thai Dika Court (in Thai), The National Conference on Information Technology 2006 (NCIT2006), Bangkok, Thailand, pp.179-186.
4) Borges, F., Bourcier, D., Andreewsky, E.and Borges, R. (2001), Conception of cognitive Interfaces for legal knowledge – Evolution of the JURISQUE project on the risks of avalanches, ICAIL 2001 The Eighth International Conference on Artificial Intelligence and Law.
5) Rosnay, M.D. (2003), Cognitive Interfaces for Legal Expressions Description – Application to Copyrighted Works Online Sharing and Transactions in D. Bourcier (ed.), Legal Knowledge and Information Systems. Jurix 2003: The Sixteenth Annual Conference. Amsterdam: IOS Press, 2003, pp. 121-130.
6) Justice XML Structure Task Force, The Justice XML Data Model:Overview and Status, http://it.ojp.gov/jxdm.
7) Vachanasvasti, K. (2006), Criminal Law: Part 1 (in Thai), Thammasat university Publishing.
8) Coyle, M., XmlGridControl Editing XML Data in the PropertyGrid, from http://www.codeproject.com/KB/grid/XmlGridControl.aspx
9) Likasitwattanakul, S. (2008), Criminal Law Code (in Thai), Winyuchon Publishing.