


Article

# A Sentence Prediction Approach Incorporating Trial Logic Based on Abductive Learning

Long Ouyang , Ruizhang Huang, Yanping Chen and Yongbin Qin \*

State Key Laboratory of Public Big Data, College of Computer Science and Technology, Guizhou University, Guiyang 550025, China

\* Correspondence: ybqin@foxmail.com

**Abstract:** Sentencing prediction is an important direction of artificial intelligence applied to the judicial field. The purpose is to predict the trial sentence for the case based on the description of the case in the adjudication documents. Traditional methods mainly use neural networks exclusively, which are trained on a large amount of data to encode textual information and then directly regress or classify out the sentence. This shows that machine learning methods are effective, but are extremely dependent on the amount of data. We found that there is still external knowledge such as laws and regulations that are not used. Moreover, the prediction of sentences in these methods does not fit well with the trial process. Thus, we propose a sentence prediction method that incorporates trial logic based on abductive learning, called SPITL. The logic of the trial is reflected in two aspects: one is that the process of sentence prediction is more in line with the logic of the trial, and the other is that external knowledge, such as legal texts, is utilized in the process of sentence prediction. Specifically, we establish a legal knowledge base for the characteristics of theft cases, translating relevant laws and legal interpretations into first-order logic. At the same time, we designed the process of sentence prediction according to the trial process by dividing it into key circumstance element identification and sentence calculation. We fused the legal knowledge base as weakly supervised information into a neural network through the combination of logical inference and machine learning. Furthermore, a sentencing calculation method that is more consistent with the sentencing rules is proposed with reference to the Sentencing Guidelines. Under the condition of the same training data, the effect of this model in the experiment of responding to the legal documents of theft cases was improved compared with state-of-the-art models without domain knowledge. The results are not only more accurate as a sentencing aid in the judicial trial process, but also more explanatory.

**Keywords:** machine learning; judicial sentencing; abductive learning; artificial intelligence



**Citation:** Ouyang, L.; Huang, R.; Chen, Y.; Qin, Y. A Sentence Prediction Approach Incorporating Trial Logic Based on Abductive Learning. *Appl. Sci.* **2022**, *12*, 7982. <https://doi.org/10.3390/app12167982>

Academic Editor: Dariusz Mazurkiewicz

Received: 28 June 2022

Accepted: 5 August 2022

Published: 9 August 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the construction of the rule of law, the large number of cases makes the court overburdened, and the time required to train highly qualified judges makes it impossible to fundamentally solve the contradiction of having too many cases and too few judges by simply increasing trial resources and increasing the number of judges. Therefore, we can use sentencing prediction techniques to reduce the burden on judges. Sentencing prediction is an important aspect of intelligent justice trials that aim to use computer technology to predict sentences from semi-structured adjudication documents.

As shown in Figure 1, the adjudication documents are mainly composed of the court case number, name of the court, defendant information, description of the case, judgment elements, legal articles involved, and trial results.

(2013)x刑初字第xx号	Case number	(2013)The x Criminal Case No. xx
贵州省贵阳市南明区人民法院刑事判决书 (2015) 南刑初字第xx号公诉机关贵阳市南明区人民检察院	Name of court	Criminal judgment of Nanming District People's Court, Guiyang city, Guizhou Province(2015)Public Prosecution No. xx Guiyang Nanming District People's Procuratorate.
被告人吴x, 曾用名吴x, 男, 1997年4月5日出生于...初中文化, 住贵州省...	Defendant information	Defendant Wu X, formerly known as Wu X, male, born on April 5, 1997... Junior high school education, guizhou Province...
经审理查明, 2013年4月5日晚上, 被告人吴x在...出租屋内盗走黑色笔记本电脑一台...第二天将电脑变卖, 得700元...	Accident description	After the trial, it was found that on the evening of April 5, 2013, the defendant Wu X ... the theft of a black laptop from a rented apartment ... the next day he sold the computer and got 700 yuan...
本院认为, 被告人吴x, 盗窃他人财物, 价值人民币10000元, 其行为已构成盗窃罪...盗窃数额为较大...行为应认定为坦白...	Judgment elements	This court finds that defendant Wu X ... stealing other people's property, the value of 10,000 yuan., its behavior has constituted the crime of larceny ... the amount of theft is larger... conduct should be considered confessional...
据此, 依照《中华人民共和国刑法》第二百六十四条、第六十七条、第五十二条、第五十三条、第六十四条之规定,	Legal articles involved	Accordingly, in accordance with articles 264, 67, 52, 53 and 64 of the Criminal Law of the People's Republic of China,
判决如下: 被告人吴x犯盗窃罪, 判处有期徒刑八个月, 并处罚金人民币一千元。	Trial results	The verdict is as follows: the defendant, Wu X, was sentenced to eight months in prison and fined 1,000 yuan.

**Figure 1.** Adjudication documents’ structure. On the left is the structure of Chinese adjudication documents, and on the right is its English equivalent.

Among the present methods, there are many that use neural networks for sentencing prediction. This can be roughly divided into three different solutions: first, feature extraction of the case description part in the adjudication document; second, regression prediction by connecting a layer of the neural network directly afterwards; third, dividing the sentence into intervals to transform it into a classification problem. However, these strategies only use the case description part of the decision document without using external knowledge such as the law, and these rely on the lack of interpretation of a large amount of data. Huang [1] uses inverse abductive learning, which combines machine learning and logical reasoning, and uses the relationships summarized in the data as external knowledge to solve the problem of a small amount of labeled data, called semi-supervised abductive learning (SS-ABL); yet, it still lacks the explanatory and logical aspects of the trial. Traditional methods use a neural network to directly fit the final sentence, which does not fit the logic and flow of the trial.

Thus, the traditional methods of sentence prediction have the following problems: they do not make full use of the external knowledge of legal features; they do not refer to the process of the judge’s trial, and the reference basis of the trial is not clear; the method of judging the sentence is not in line with the process and logic. We conducted the following research to address these issues.

We noticed in this particular area of law that there is some logical and rule-based knowledge, such as legal texts and judicial interpretations. These rules can be used as a priori knowledge and act as a weak supervisor in sentencing prediction, which is not used in the traditional approach, such as: (1) If the clause about intentional homicide appears in the adjudication document, the circumstance of intentional homicide must have appeared in the case of the offender, and then, the sentence will be increased at the trial. (2) It is clearly stated in the law that different degrees of crime correspond to different levels of sentencing. For instance, when the amount of theft is smaller, the sentence in the law will be a shorter range. When the amount is larger, the sentence is in a longer range. This makes different amounts of theft under the provisions of the law affect the base sentence, and thus the final sentence. Thus, we note that there is a certain logical nature between key circumstance elements, legal provisions, and legal interpretations. This nature can be applied to logical reasoning, allowing it to play a supervisory role in both the identification of key circumstance elements and the calculation of the sentence. If the result of the identification does not conform to the logic in the law, one of the results can be changed to make it conform to that logic to the maximum extent possible. This allows for explanatory correction of misidentified labels and improves recognition rates.

First of all, in order to use this external knowledge, we write the external knowledge in the form of first-order logic to form a logical knowledge base for logical reasoning. Next, we took a theft case as an example and extracted 23 key trial circumstance elements by referring to the legal provisions, as well as the “Guidelines of the Supreme People’s Court on Sentencing for Common Crimes” (“Sentencing Guidelines”). These key plot elements are highly informative for trial purposes. Additionally, to design a sentence calculation that is logical for the trial, we refer to the Sentencing Guidelines. There are four main features

as follows: (1) the basis for sentencing is the key circumstantial element of the criminal act of the basic crime-constituting facts; (2) the starting point of the sentence is ruled within the range of the sentence in accordance with the provisions of the law; (3) increase or decrease the starting point of sentencing based on other criminal facts affecting the amount, number, consequences, and purpose of the crime; (4) consider the entire case and decide the final sentence to be pronounced. This shows that sentencing needs to be based not only on the basic constituent facts of the case, i.e., the key elements of the case, but also on the totality of the circumstances of the case. In determining the base sentencing, it is necessary to refer to the range of sentences stipulated in the legal provisions.

Therefore, combining the above studies and findings to address the lack of trial logic, inadequate use of knowledge of the law, and lack of interpretability in the traditional approach, we propose a new model based on the existing algorithm SS-ABL. The main innovation points are as follows:

- Prepared and expanded the legal knowledge base (KB). Not only does it use common sense constraints from data, but it also converts legal texts and judicial interpretations into a form of first-order logic to constitute the knowledge base.
- Applying knowledge-base-based logical reasoning to the process of key circumstance element identification and sentencing calculation makes it more adequate for both the upstream and downstream tasks of sentencing prediction to provide some oversight.
- Incorporating the descriptive features of the case and key circumstance elements, a sentencing calculation method that is more in line with the judge's trial process was developed in accordance with the Sentencing Guidelines.

The paper is structured as follows. Relevant work in this area is described in Section 2. Section 3 details the sentencing prediction model that combines domain knowledge. The evaluation and results are presented in Section 4. The conclusions of the paper are presented in Section 5.

## 2. Related Work

Case sentencing prediction has been studied by domestic and foreign researchers for many years [2–5]. Since the last century, researchers have been exploring the possibility of using mathematical methods to quantitatively analyze and predict the behavior of justice. Kort [6] analyzes a number of decided cases to determine the factual elements that affect the verdict, scores those elements using a mathematical formula, and uses the resulting content to aid the judge's decision. Shapira [7] implemented a system for juvenile probation decision-making for probation officers in Israel using a rule-based expert system [8]. Sulea et al. [9] used an integrated model of multiple support vector machines (SVMs) to predict the outcome of French Supreme Court cases based on a large-scale corpus of cases. Liu and Hsieh et al. [10] proposed extracting shallow textual features (e.g., characters, words, and phrases) from adjudication documents for case charge prediction. Katz et al. [11] predicted the outcome of a U.S. Supreme Court decision by using random forest (RF) to extract valid features from the case description.

In early studies, researchers mostly conducted research on sentence prediction by means of rules, mathematical statistics, and machine learning. However, the development of research on automated sentence prediction methods has not been long.

With the development of deep learning techniques, most researchers have studied the task of sentencing prediction research on the idea of adopting text classification [12] by extracting some features. The Challenge of AI in Law 2018 (CAIL2018) and Legal AI Challenge 2021 (LAIC2021) both opened sentencing prediction tracks, and the baseline model used was the traditional text classification method. This approach transforms the sentencing prediction problem into a text classification problem by dividing the sentencing into intervals. Luo et al. [13] proposed a neural network model based on an attention mechanism to predict case charges based on the factual description of the case by introducing relevant legal provisions. Hu et al. [14] blend 10 different legal attributes to predict a small number of shooting and confusion charges. Ye et al. [15] proposed a sequence-to-

sequence (Seq2Seq) model under labeling conditions to provide a courtroom view to assist in sentencing based on factual descriptions of criminal cases with coding of charge labels. Wu et al. [16] study the sentencing of environmental rights cases from the perspective of international criminal law and uses convolutional neural networks (CNNs) to determine the sentencing of environmental rights cases. Their results show that the introduction of CNNs improves the effect of the sentencing term prediction model and the fine prediction model significantly. Yang et al. [17] proposed a hierarchical attention network (HAN) that combines static spatial information, short-term motion information, and a long-term video temporal structure for complex human behavior understanding, while Wang et al. [18] later applied HAN to sentencing prediction models by using residual networks to fuse an improved hierarchical attention network (iHAN) and a deep pyramidal convolutional neural network (DPCNN) and proposed a hybrid deep neural network model, hybrid attention and CNN (HAC), to apply HAC to sentencing prediction models. Park et al. [19] used a multi-task deep learning model with an attention mechanism for combining three tasks (accident types, applied articles, and the sentencing of ship accidents) for sentencing prediction. All of the above methods use text classification methods for sentencing prediction. Zhong et al. [20] considered the existence of dependencies between multiple subtasks of legal trials (e.g., applicable legal provisions, charges, fines, and sentencings) and proposed a multitask topological dependency learning model called TOPJUDGE to predict the legal trials of cases. Yang et al. [21] proposed a multiperspective bidirectional feedback network (MPBFN) based on topological dependencies between multiple subtasks and the introduction of combinatorial semantic relations between words to predict the legal trials of cases. Zhou et al. [22] proposed an inverse deduction learning framework to solve the problem that machine learning and logical reasoning are difficult to combine by introducing a knowledge base in the logical reasoning part and using a heuristic trial-and-error search algorithm to combine machine learning. Huang et al. [1] used the inverse deduction learning framework to solve the problem of case element identification by taking advantage of the strong logic of the joint trial documents.

The current research on sentencing prediction in general can be divided into two methods: text classification and regression analysis prediction. Text classification methods can be combined with crime prediction and law recommendation to achieve better results. In contrast, regression analysis prediction identifies the elements of the crime or the relevant law for prediction. The textual classification method may focus on the semantic information of the crime elements, but ignores the final calculation of the sentencing, and the regression analysis method focuses on the construction of the regression model, but does not pay attention to the information of the key crime circumstance elements themselves. The above methods do not combine the characteristics of the other methods well, so they do not achieve better results in sentencing prediction.

### 3. Methods

In this section, we will describe our proposed model in terms of the logic of the trial: it is divided into three modules, namely knowledge base preparation, element identification, and sentencing calculation. This sentence prediction model is an effective combination, and it is a new contribution of our work. In the following, we will present each of these three components.

#### 3.1. Knowledge Base

When we combed through the statutes and adjudication documents, we sorted out approximately 23 key circumstance elements that can also be called trial elements to better use the logical relationships between the statutes, between the statutes and adjudication documents, and within the adjudication documents, referring to the Sentencing Guidelines. See Table 1.

**Table 1.** The key circumstance elements.

Element	Ratio	Element	Ratio
In Hospital	2.56	Minor	10.96
Production Materials	0.22	Burglary	21.56
Returned Items	1.47	Pickpocket	14.42
Main Culprit	2.59	Accomplice	2.62
Carrying Weapons	1.39	Surrendering	10.61
For Drug Addiction	1.94	First Offender	6.19
For Gambling	0.25	Many Times	35.11
Psychiatric Patients	0.11	Have Criminal Record	19.19
Compensation for Damages	57.17	Plead Guilty	35.20
Seniors	0.16	Deaf and Mute	1.06
Blind Person	0.05	Get Forgiveness	8.04
Confession	62.38		

Among these are the key circumstance elements, which are elements summarized by the circumstances that have a significant impact on the sentence according to the Sentencing Guidelines. The ratio refers to the number of occurrences of the element as a percentage of the number of occurrences of all elements. With these key elements, we followed this table to extract the corresponding elements in the law, such as theft, multiple theft, etc., and the corresponding sentence ranges. Then, we chose to represent their logical structure using first-order logical relations to form a knowledge base. At the same time, we observed some potential logical relationships among the key plot elements in the data of the adjudication documents when we organized them. As you can see, the legal knowledge base is composed of first-order logical forms bounded by legal rules and common sense. The key plot elements involved in each judgment document are also present in the legal knowledge base in a first-order logical form. A portion of the legal knowledge base is shown in Figure 2.

刑法第264条 【盗窃罪】 1. 盗窃公私财物，数额较大或者多次盗窃的，处三年以下有期徒刑、拘役或者管制，并处或者单处罚金； 2. 数额巨大或者有其他严重情节的，处三年以上十年以下有期徒刑，并处罚金；...	Article 264 of the Penal Code [Theft] 1. Whoever steals public or private property in a relatively large amount or repeatedly shall be sentenced to fixed-term imprisonment of not more than three years, criminal detention or public surveillance and shall also, or shall only, be fined; 2. If the amount involved is huge, or if there are other serious circumstances, he shall be sentenced to fixed-term imprisonment of not less than three years but not more than 10 years and shall also be fined;...
--	--

(a)

law (264, 1) → Amount_stolen (Money, larger_amount) ∨ element(many_times)
law (264, 1) → score(0, 36)
False ← element(pickpocket) ∧ element(burglary)

(b)

**Figure 2.** Examples of laws and part of the legal knowledge base. (a) Part of Article 264 of the Criminal Law of the People’s Republic of China. (b) Part of the legal knowledge base.

This figure shows how we parse the law and transform it into a first-order logical form. In Figure 2a, on the left is Article 264 of the Chinese Criminal Law and on the right is its corresponding English version. The blue section is the key circumstance elements, and the red section is the range of sentencing given. In Figure 2b, the part in the box

above corresponds to the result of transforming the part into first-order logic where the “score” value is the sentence interval specified in the law. The following part is an example of the common sense constraint. In the legal interpretation document, “pickpocket” is defined as “stealing property carried by another person in a public place or on public transportation.” and “burglary” is defined as “entering a room and stealing it secretly.” Obviously, “pickpocket” and “burglary” do not exist at the same time. The legal rules are decomposed into sentence intervals corresponding to key circumstance elements through legal provisions and judicial interpretations, which have reference values for subsequent sentencing calculation. For example, in Figure 2b, if the first paragraph of section 264 is met, the “score”, which means the range of the sentence, is between 0 and 36 months (sentences are in months in this article). The common sense constraint describes the relationship between key plot elements and has a weak supervisory effect on the subsequent identification of key plot elements.

### 3.2. Key Circumstance Element Identification

In this section, we use a method based on the abductive learning method to identify key plot elements, the flow of which is shown in Figure 3.

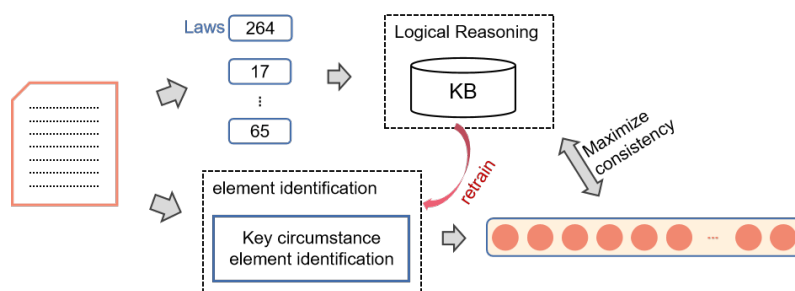


Figure 3. The architecture of key circumstance element identification.

In the first part, we identified the key plot descriptions in the adjudication documents by the element extraction algorithm. To better obtain the key plot elements of the case and to solve the problem of the insufficient generalization of existing methods, we adopted the BERT [23] pretraining model for the extraction of text features through the comparison of multimodel experiments. The model employs a connection using the bidirectional encoder block layer part of the transformer [24] model, discarding the decoder module so that it automatically has bidirectional coding capability and powerful feature extraction capability. Transformer’s encoding sequence uses the idea of a self-attention mechanism that can read the whole text sequence at once. By capturing the global contextual information to establish a long-range dependency on the target, stronger features of the whole text can be extracted.

First, we need to process the text descriptions in the key episodes into the standard input form of BERT. We add the marker “[CLS]” as the start of a sentence and the marker “[SEP]” as the end of a sentence to the entire text. The initial input is  $W = \{W_1, W_2, \dots, W_n\}$ , where  $W_i$  is a word in a sentence and  $n$  is the length of the sentence from each sentence in the judgment elements section. Then, three kinds of embeddings corresponding to  $W$  are added together to obtain the BERT input:

$$E_{input} = E_t + E_s + E_p \tag{1}$$

where  $E_t$ ,  $E_s$ ,  $E_p$  are the token embeddings, segment embeddings, and position embeddings corresponding to  $W$ , respectively. Since this paper is a single-sentence task, the identity vector here is 0, and the inclusion of the position vector considers that the sequential nature of the input sequence has an effect on the information in the text. The final input of BERT is denoted as  $E_{input} \in R^{n \times d}$ , where  $n$  denotes the length of the input text and  $d$  denotes the dimension of the vector.

A bidirectional transformer encoder block is used for connection in BERT, discarding the decoder module so that it automatically has bidirectional encoding capability and powerful feature extraction capability. In the output of BERT, for each input word embedding, there is a corresponding output, and since the first token of each output sequence is always a special classification, i.e., “[CLS]” token embedding, which does not contain any semantic information, the vector is a collection of information of a whole sentence. We used this vector to identify the key plot elements. It is followed by a fully connected layer that transforms the vector dimension into the length of the number of key plot elements ( $m$ ) and by sigmoid as the activation function:

$$P(x) = \text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

The probabilities of the key element classification are normalized and transformed into a probability distribution with a 0 to 1 interval. We set a threshold value:

$$y_p(x) = \begin{cases} 0, & P(x) < \alpha \\ 1, & P(x) \geq \alpha \end{cases} \quad (3)$$

and the corresponding key plot element exists when its probability exceeds that value  $\alpha$ . Then, we obtain the temporary key plot elements  $y_p$ , which can also be called pseudo-labels.

In the second part, we used regular expressions to extract from the involved law articles in the adjudication documents. Additionally, we used the pseudo-labels obtained by element identification as the input. The results of their reasoning generated by the legal knowledge base (as a set) are corrected for the provisional key elements by maximizing the consistency of operation to obtain the most likely result as the final key plot element. The maximizing consistency is that after inputting the temporary key elements, the logical reasoning of the legal knowledge base will produce some corrected results that conform to the logical rules, and we only need to select one of them in the end. The basis of selection is to pass the corrected list of key elements through the sentencing calculation module to produce the corresponding sentencing loss and select the one with the smallest loss as the final key circumstance element. Finally, we obtain the corrected label for the pseudo-labels.

The whole algorithmic process of key circumstance element identification presented in this section is shown in Algorithm 1.

---

**Algorithm 1** Key circumstance element identification.

---

**Input:** Judgment elements' data  $X_i$  of judgment document with their labels  $Y_i$ ; unlabeled data  $X_j$  with their legal articles involved  $L_j$  and true length of sentence  $S_j$ ; legal knowledge base (KB)

**Output:**  $X_j$  corresponding to label  $Y_j$

```

1:  $f \leftarrow \text{TrainModel}(X_i, Y_i)$ 
2:  $Y_j \leftarrow f(X_j)$  # Generate pseudo-labels
3:  $Laws \leftarrow \text{RegularExpressions}(L_j)$ 
4: while  $t$  is change do
5:    $\Delta(Y_j) \leftarrow \text{Abduce}(\text{KB}, Laws, Y_j)$  # Revise pseudo-labels
6:    $\hat{t} \leftarrow \text{SentencingCalculation}(\Delta(Y_j))$ 
7:    $t = \text{Min}(|\hat{t} - S_j|)$ 
8: end while
9:  $f \leftarrow \text{ReTrainModel}(f, X_j, Y_j, X_i, Y_i)$  # Update recognizer  $f$ 
10: return  $Y_j$ 

```

---

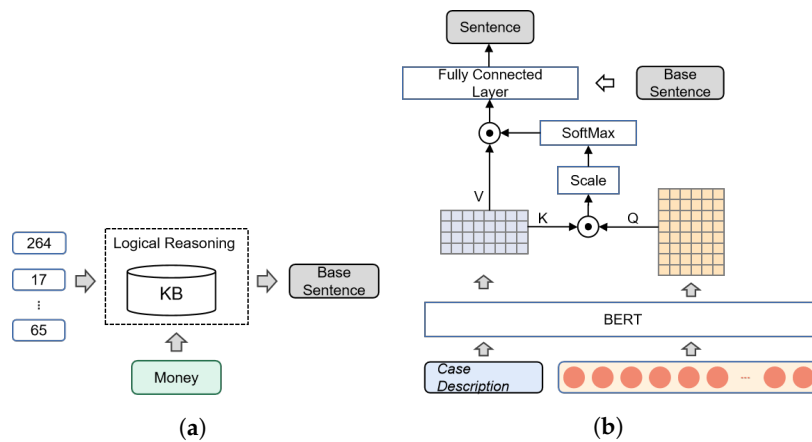
### 3.3. Sentencing Calculation

To be more in line with the flow of the trial, the sentence calculation module was designed in accordance with the Sentencing Guidelines. As shown in Figure 4, we can see

the process for obtaining a base sentence and a final sentence separately. We first combined the amount of money involved in the case obtained from the judgment documents with the sentence range obtained by logical reasoning based on the legal knowledge base to design a method to obtain the base sentence.

$$BaseSentence = \beta m + \gamma \tag{4}$$

where  $\beta$  represents the adjustment of the amount of theft to the starting sentence,  $m$  represents the amount of theft in the case, and  $\gamma$  represents the starting sentence in the statute. Since the base sentence is not defined in detail in the Sentencing Guidelines, we can only calculate it based on one of the reference theft amounts, while the base sentence can only be within the range specified in the involved law. Therefore, we constructed a one-dimensional function of the amount to calculate the base sentence.



**Figure 4.** A flow chart of the trial according to the relevant documents. (a) Base sentence; (b) final sentence.

To focus on the key plot elements and consider the global features of the case, we combined the key plot elements with the general features of the case by referring to the self-attention [24] approach. In this way, not only the global features, but also the key circumstances are considered. By combining the information of key plot elements and general features of the case, we generated an attention weight for each part of the text and weighted the text information, which can help us better capture the information related to the key plot in the text paragraph.

We first performed feature extraction on the two texts by BERT as in Section 3.2 and obtained the vector  $D$  of common features of the case and the feature vector  $E$  of key elements of the case, where both  $D$  and  $E$  are global information about the respective text.

Then, we put these two vectors through a control attention mechanism to obtain features about the plot descriptions that exacerbate the critical plot element parts.

$$V = W^V D \tag{5}$$

$$K = W^K D \tag{6}$$

$$Q = W^Q E \tag{7}$$

multiplying the two eigenvectors with the matrix  $W$  yields  $V$ ,  $K$ , and  $Q$ .

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{8}$$

We calculated the dot product between  $Q$  and  $K$ . Next, for its result to be too large, a constraint operation was performed to divide the result by  $\sqrt{d_k}$ , where  $d_k$  is the dimensionality of the vector  $K$ . After that, a softmax layer was added to it to normalize the result



and then multiplied by the matrix  $V$  to obtain the final weight summation representation *Attention*. Moreover, the obtained results were plugged into a layer of the neural network, while the base sentence was added for linear regression, and finally, the final sentence was obtained.

The whole algorithmic process of sentencing calculation presented in this section is shown in Algorithm 2.

---

**Algorithm 2** Sentencing calculation.

---

**Input:** Accident description data  $D_i$  of judgment document and amount of theft  $M_i$ ; key circumstance element  $Y_i$ ; real sentence  $S_i$ ; unlabeled data  $D_j$  with their related laws *Laws*, amount of theft  $M_j$  and key circumstance element  $Y_j$  obtained from a previous mission; legal knowledge base (KB)

**Output:**  $D_j$  corresponding to sentence  $S_j$

- 1:  $g \leftarrow \text{TrainModel}(D_i, M_i, Y_i, S_i)$  # Training neural network  $g$
  - 2:  $\gamma \leftarrow \text{Abduce}(M_j, \text{Laws}, \text{KB})$
  - 3:  $\text{BaseSentence} \leftarrow \beta \times M_j + \gamma$
  - 4:  $S_j \leftarrow g(\text{BaseSentence}, D_j, Y_j)$
  - 5: **return**  $S_j$
- 

#### 4. Experiments and Results

In this section, we carried out experiments on the theft judicial sentencing task to demonstrate that SPITL is able to leverage unlabeled data and symbolic knowledge.

##### 4.1. Experimental Setup and Evaluation Index

Since there is no public dataset on evidence extraction from judgment documents, we obtained the relevant judgment documents from the People's Court of Guizhou Province, erased the sensitive information, and used manual annotation to construct the dataset. The dataset contains 3668 adjudication documents of theft cases, and the training sets with 10%, 50%, and 90% of the cases were used for the experiments. To better measure the performance of the model, we adopted the Monte Carlo cross-validation method by randomly dividing the dataset into training and validation sets each time, so that three-time separate model training and validations were performed, and finally, these validation results were averaged as the validation error of this model. We set the length of the input text to 512, the size of the hidden layer to 768, the threshold  $\alpha$  to 0.25,  $\beta$  in the base penalty to 0.065, and  $\gamma$  to the minimal sentence, where  $\beta$  is mainly used to reduce the amount of integers to decimals in order to facilitate the calculation by the neural network and  $\gamma$  is used to obtain the starting penalty specified in different laws through logical reasoning. In particular, the effect of the  $\alpha$  value on the results is relatively large, and we will analyze its effect in Section 4.4. Experiments ran on a workstation with AMD EPYC 7742 64-Core Processor, 40 GB memory, and the A100-SXM4-40GB GPU.

We chose to use the mean absolute error (MAE), mean-squared error (MSE), root-mean-squared error (RMSE), and R2Score (coefficient of determination  $R^2$ ) as the evaluation criteria. The MAE is a loss function used in regression models and is the sum of the absolute values of the differences between the target and predicted values, which measures the average modal length of the error between the predicted and target values. It can better reflect the actual situation of prediction value error. The formula is as follows, where  $y$  is the true value and  $\hat{y}$  is the predicted value:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n | \hat{y}_i - y_i | \quad (9)$$

the MSE is the sum of squares between the predicted and true values, which is slightly simpler to calculate than the MAE, but has strong robustness. It can be used to measure

the dispersion of the difference between the predicted and true values. The formula is as follows, where  $y$  is the true value and  $\hat{y}$  is the predicted value:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (10)$$

The RMSE is the square root of the ratio of the sum of squares of deviations of observations from the true value to the number of observations  $n$ . It is used to measure the deviation of the observed value from the true value and is often used as a measure of the prediction results of machine learning models. The formula is as follows, where  $y$  is the true value and  $\hat{y}$  is the predicted value:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (11)$$

The R2Score is generally used in regression models to assess the degree of conformity between predicted and actual values and is defined as follows:

$$\text{R2Score} = 1 - FVU = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (12)$$

where  $y$  is the true value,  $\hat{y}$  is the predicted value,  $\bar{y}$  is the average of the true values,  $FVU$  is called the fraction of variance unexplained,  $RSS$  is called the residual sum of squares, and  $TSS$  is called the total sum of squares. In general, the closer R2Score is to 1, the better the model fitting effect.

#### 4.2. Result Analysis

We chose five baseline models for the comparison experiments:

- **TFIDF-SVM:** We transformed sentence prediction into a classification problem by dividing the sentence intervals into one class. The text is first divided into words, then feature extraction is performed using TFIDF, classification is performed using SVM [25], and the median of the intervals is used as the prediction result.
- **CNN [26]:** Using a CNN-based classification model with multiple filter window widths for text classification, we transformed sentence prediction into a classification problem by taking the final sentence as the middle value of the classification interval.
- **LSTM-FC [27]:** A vectorized representation of the text using LSTM is followed by a linear layer for sentencing prediction.
- **BERT-FC [23]:** Feature extraction is performed on the text using BERT, and sentence prediction is performed in a linear layer.
- **SS-ABL [1]:** Sentence prediction directly using the inverse abductive learning framework.

Comparative experiments are shown in Table 2.

We can see that comparing the three models with different training set proportions, our model is better in terms of key element recognition and sentence calculation. Moreover, when the training set is 90%, the error of our model in the prediction of the prison sentence is within three months, reaching 2.6 months. Although the MAE of the SS-ABL model is only slightly lower than that of our model when the training set accounts for 10%, the MSE and RMSE are higher and show that the prediction result fluctuates greatly and does not have good stability. Again, we can see from the R2Score that the SPITL model is better. For the R2Score, the closer to 1, the better the fit of the model is and the better the model is. It can also be seen from the table that the R2Score of the SPITL model is always closer to 1 than those of the other models for the same training ratio. Especially when the training accounts for a relatively small amount of time, the difference with other models is more obvious, which should be played by the use of external knowledge through logical reasoning.

**Table 2.** Performance results for different methods.

Model	Training Ratio	MAE	MSE	RMSE	R2Score
TFIDF-SVM	10%	8.50	89.06	9.44	0.3772
CNN	10%	8.04	80.47	8.97	0.4286
LSTM-FC	10%	7.95	80.28	8.97	0.4173
Text CNN	10%	8.34	82.74	9.10	0.3858
BERT-FC	10%	7.12	80.39	8.96	0.4592
SS-ABL	10%	<b>3.46</b>	56.48	7.52	0.5934
SPITL	10%	3.49	<b>41.55</b>	<b>6.45</b>	<b>0.6684</b>
TFIDF-SVM	50%	7.50	78.35	8.85	0.4592
CNN	50%	7.04	80.47	8.97	0.4257
LSTM-FC	50%	6.34	82.74	9.10	0.3876
Text CNN	50%	7.34	82.74	9.10	0.3928
BERT-FC	50%	5.68	67.20	8.20	0.6735
SS-ABL	50%	3.53	45.36	6.73	0.6472
SPITL	50%	<b>3.25</b>	<b>34.94</b>	<b>5.91</b>	<b>0.7895</b>
TFIDF-SVM	90%	7.02	70.33	8.39	0.6782
CNN	90%	5.86	64.36	8.02	0.6895
LSTM-FC	90%	5.32	52.71	7.26	0.6361
Text CNN	90%	5.51	49.56	7.03	0.6530
BERT-FC	90%	4.25	40.42	6.36	0.6982
SS-ABL	90%	3.28	38.05	6.17	0.7468
SPITL	90%	<b>2.62</b>	<b>22.83</b>	<b>4.77</b>	<b>0.8734</b>

From the above experiments, it is clear that SPITL outperforms the traditional model. Since most traditional models are based on using data to adjust parameters to fit the results, the accuracy is not particularly high when the amount of data is small. However, this model combines machine learning and logical reasoning and uses logical reasoning to constrain and change the results of recognition when the amount of data is small, resulting in a higher recognition rate and a smaller error in the sentence prediction. For example, when it is known that the case involves Article 17 of the criminal code (About minors), then the key circumstance element must contain the “Minor”, and if the identified pseudo-label does not contain that label, it is modified to contain that label.

In addition, we verified the validity of the sentence calculation method. As shown in Table 3, when key plot elements were obtained, we used different calculation methods for comparison. This means that we used different methods for sentence calculation when the correct key elements are known.

**Table 3.** Calculation of sentences for different methods.

Model	Training Ratio	MAE	MSE	RMSE	R2Score
SVM	50%	3.43	59.96	7.74	0.5853
KNN	50%	3.23	38.44	6.20	0.6893
Bayes	50%	7.32	36.46	6.03	0.7239
Decision Tree	50%	3.58	41.93	6.48	0.6702
Linear Regression	50%	5.02	68.15	8.26	0.4570
Our Model	50%	<b>3.01</b>	<b>32.72</b>	<b>5.72</b>	<b>0.7864</b>
SVM	90%	3.04	26.55	5.15	0.8075
KNN	90%	3.04	23.69	4.87	0.8210
Bayes	90%	7.11	30.03	5.48	0.7926
Decision Tree	90%	3.24	29.41	5.42	0.8083
Linear Regression	90%	3.53	29.11	5.40	0.8095
Our Model	90%	<b>2.60</b>	<b>20.76</b>	<b>4.56</b>	<b>0.8962</b>

We can see that comparing some of the common methods of calculating sentences, our calculation method is superior to the other methods. When having more training sets,

the MAE, MSE, and RMSE values are smallest, and obviously, our model is more accurate and stable. Furthermore, at the same training ratio, our model has a higher R2Score value, closer to 1. From this perspective, the sentence calculation method of our model is better than the others.

With the increase in the amount of data, the accuracy of traditional identification improved, but in the calculation of sentences, it cannot absolutely rely on the key circumstance elements, but needs to refer to the whole process of the case, such as the motive for the crime, the means of the crime, etc. Therefore, the new sentence calculation model we used will be more accurate with less error.

#### 4.3. Chi-Squared Test

In order to check whether there is a significant difference between the actual ( $O$ ) and expected results ( $E$ ), we used the mathematical statistics method of the chi-squared test to perform hypothesis testing. First, we established the test hypothesis and determined the test level.

$$\begin{aligned} H_0 &: O = E \\ H_1 &: O \neq E \end{aligned}$$

Meanwhile, we set the significance level  $\alpha$  to 0.05. We calculated the cardinality ( $\chi^2$ ) by predicting the sentence and the true sentence.

$$\chi^2 = \sum_{i=1}^n \frac{(O_i - E_i)^2}{E_i} \quad (13)$$

We used the SPITL results after the 90% training set to perform the operation and obtained a  $\chi^2$  value of 340.8386. Since we know that its degree of freedom is the number of data minus one, i.e.,  $(n-1)$ , we can obtain a right-tailed probability of cardinality ( $p$ -value  $P$ ) of 0.8231. Obviously,

$$P = 0.8231 > \alpha = 0.05$$

so we accept hypothesis  $H_0$ . From the above, it can be seen that the fit of the predicted values with our method to the true values is high. It also proves that the method of this paper is credible, valid, and accurate. To see more intuitively the fit between the predicted and true values, we plotted them using a graph, as shown in Figure 5, where acc is the actual value and pre is the predicted value.

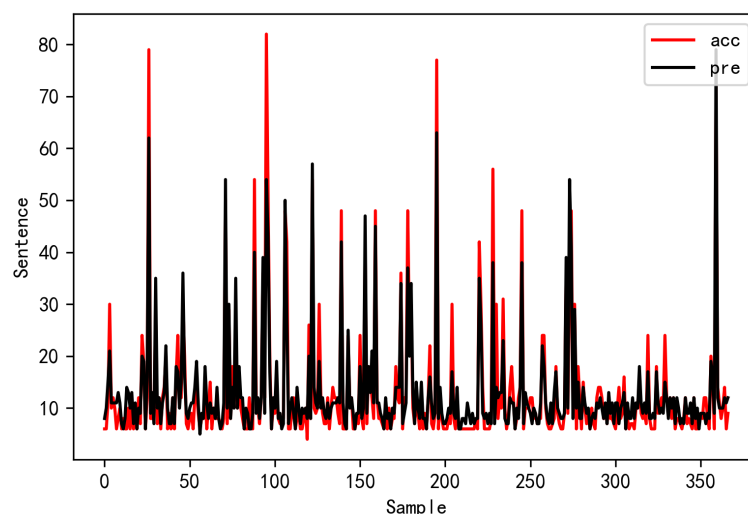


Figure 5. Line graph of actual and predicted values.

#### 4.4. Hyperparameter $\alpha$

In the model,  $\alpha$  is used as a threshold in the identification of key plot elements and has an impact on the results of identification. Therefore, we analyzed the effect of the choice of  $\alpha$  values on the recognition results when key plot elements were identified. Again, the experiments were compared at a training ratio of 90% of the settings. Referring to Figure 6, we can see that different  $\alpha$  values still have an impact on the accuracy of key plot elements' recognition. At 0.25, the F1-value of recognition is the highest.

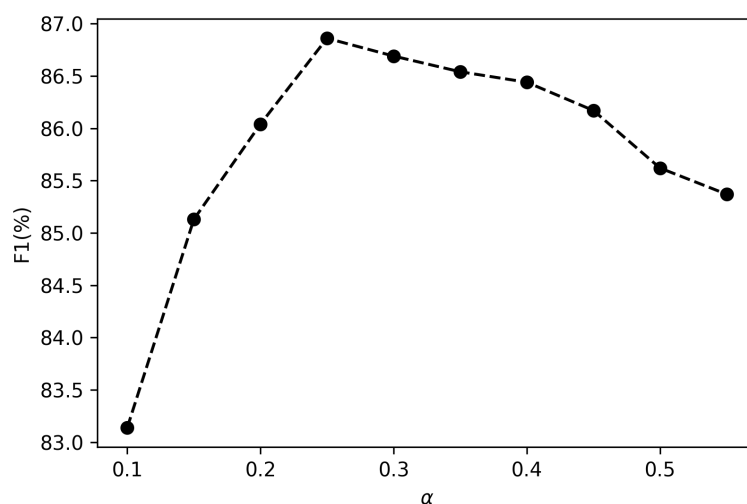


Figure 6. Line graph of  $\alpha$  and element identification F1-value.

## 5. Conclusions

In this paper, we proposed a sentencing prediction model framework called SPITL. In this framework, our core innovations were the first-order logicalization of legal texts and their legal interpretations and adding to the legal knowledge base, which makes logical reasoning more adequate to supervise machine learning, and the design of a sentence calculation method that is more consistent with the trial process. The model is set-associative and consists of two subtasks, key circumstance element identification and sentence calculation. It was experimentally demonstrated that this model not only solves the problem of few labeled data by an abductive learning framework in sentence prediction, but is also more accurate and explanatory in sentence calculation. Despite our model achieving some results on theft cases, it is still challenging to migrate to other cases of different categories. Since different cases have different focuses, there will be more laws involved, and the reasoning will be more complicated and may take longer. In our future work, we will work on finding a better way to integrate external knowledge into the neural network training process or designing an objective function to improve inference efficiency and shorten inference time.

**Author Contributions:** Conceptualization, L.O. and R.H.; methodology, L.O. and R.H.; software, L.O.; validation, L.O.; formal analysis, L.O.; investigation, L.O. and R.H.; resources, R.H. and Y.Q.; data curation, L.O.; writing—original draft preparation, L.O.; writing—review and editing, R.H.; supervision, R.H., Y.C. and Y.Q.; project administration, R.H. and Y.Q.; funding acquisition, Y.Q. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China under Grant No. 62066008 and Key Technology R&D Program of Guizhou Province No. [2022]277.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors would like to thank the research team members for their contributions to this work.

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

## References

- Huang, Y.X.; Dai, W.Z.; Yang, J.; Cai, L.W.; Cheng, S.F.; Huang, R.Z.; Li, Y.F.; Zhou, Z.H. Semi-Supervised Abductive Learning and Its Application to Theft Judicial Sentencing. In Proceedings of the 2020 IEEE International Conference on Data Mining (ICDM), Sorrento, Italy, 17–20 November 2020; pp. 1070–1075.
- Nagel, S.S. Applying correlation analysis to case prediction. *Tex. Law Rev.* **1964**, *42*, 1006–1017. [CrossRef]
- Keown, R. Mathematical Models For Legal Prediction. *Comput. J.* **1980**, *1*, 829–831. Available online: <https://repository.law.uic.edu/jitpl/vol2/iss1/29/> (accessed on 23 September 2021).
- Jeffrey, A.S. Predicting Supreme Court Cases Probabilistically: The Search and Seizure Cases, 1962–1981. *Am. Political Sci. Rev.* **1984**, *78*, 891–900. [CrossRef]
- Benjamin, L.; Tom, C. The Supreme Court’s Many Median Justices. *Am. Political Sci. Rev.* **2012**, *106*, 847–866. [CrossRef]
- Kort, F. Predicting Supreme Court Decisions Mathematically: A Quantitative Analysis of the “Right to Counsel” Cases. *Am. Political Sci. Rev.* **1957**, *51*, 1–12. [CrossRef]
- Shapira, M. Computerized decision technology in social service: Decision support system improves decision practice in youth probation service. *Int. J. Sociol. Soc. Policy* **1990**, *10*, 138–164. [CrossRef]
- Giarratano, J.C.; Riley, G.D. *Expert Systems: Principles and Programming*; Brooks/Cole Publishing Co.: Pacific Grove, CA, USA, 2005; ISBN 978-0-534-38447-0.
- Sulea, O.M.; Zampieri, M.; Malmasi, S.; Vela, M.; Dinu, L.P.; Genabith, J.V. Exploring the Use of Text Classification in the Legal Domain. *arXiv* **2017**, arXiv:1710.09306.
- Liu, C.L.; Hsieh, C.D. Exploring Phrase-Based Classification of Judicial Documents for Criminal Charges in Chinese. In Proceedings of the 16th International Conference on Foundations of Intelligent Systems (ISMIS), Bari, Italy, 27–29 September 2006; pp. 681–690. [CrossRef]
- Katz, D.M.; Bommarito, M.J., II; Blackman, J. A General Approach for Predicting the Behavior of the Supreme Court of the United States. *PLoS ONE* **2016**, *12*, e0174698. [CrossRef]
- Li, Q.; Peng, H.; Li, J.X.; Xia, C.Y.; Yang, R.Y.; Sun, L.C.; Yu, P.; He, L.F. A Text Classification Survey: From Shallow to Deep Learning. *arXiv* **2020**, arXiv:2008.00364v1.
- Luo, B.F.; Feng, Y.S.; Xu, J.B.; Zhang, X.; Zhao, D.Y. Learning to Predict Charges for Criminal Cases with Legal Basis. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP), Copenhagen, Denmark, 7–11 September 2017; pp. 2727–2736.
- Hu, Z.K.; Li, X.; Tu, C.C.; Liu, Z.Y.; Sun, M.S. Few-Shot Charge Prediction with Discriminative Legal Attributes. In Proceedings of the 27th International Conference on Computational Linguistics (COLING), Santa Fe, NM, USA, 20–26 August 2018; pp. 487–498.
- Ye, H.; Jiang, X.; Luo, Z.C.; Chao, W.H. Interpretable Charge Predictions for Criminal Cases: Learning to Generate Court Views from Fact Descriptions. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), New Orleans, LA, USA, 1–6 June 2018; pp. 1854–1864.
- Wu, J.X.; Wang, H.Y.; Sun, N.; Wang, H.W.; Tatarinov, D. International Criminal Law Protection of Environmental Rights and Sentencing Based on Artificial Intelligence. *J. Environ. Public Health* **2022**, *2022*, 4064135. [CrossRef] [PubMed]
- Yang, Z.C.; Yang, D.Y.; Dyer, C.; He, X.D.; Smola, A.; Hovy, E. Hierarchical Attention Networks for Document Classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), San Diego, CA, USA, 12–17 June 2016; pp. 1480–1489.
- Wang, W.G.; Chen, Y.W.; Cai, H.; Zeng, Y.N.; Yang, H.Y. Judicial document intellectual processing using hybrid deep neural networks. *J. Tsinghua Univ. Technol.* **2019**, *59*, 505–511. [CrossRef]
- Park, H.M.; Kim, J.H. Multi-Task Deep Learning Model with an Attention Mechanism for Ship Accident Sentence Prediction. *Appl. Sci.* **2022**, *12*, 233. [CrossRef]
- Zhong, H.X.; Guo, Z.P.; Tu, C.C.; Xiao, C.J.; Liu, Z.Y.; Sun, M.S. Legal Judgment Prediction via Topological Learning. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), Brussels, Belgium, 31 October–4 November 2018; pp. 3540–3549.
- Yang, W.M.; Jia, W.J.; Zhou, X.J.; Luo, Y.T. Legal Judgment Prediction via Multi-Perspective Bi-Feedback Network. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI), Macao, China, 10–16 August 2019; pp. 4085–4091.
- Zhou, Z.H. Abductive learning: Towards bridging machine learning and logical reasoning. *Sci. China Inf. Sci.* **2019**, *62*, 220–222. [CrossRef]
- Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2017**, arXiv:1706.03762.
- Zhou, C.C.; Lin, C.J. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 2157–6904. [CrossRef]

- 
26. Kim, Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1746–1751.
  27. Shi, X.J.; Chen, Z.R.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015; Volume 1, pp. 802–810.