Research Article

Learning Users' Intention of Legal Consultation through Pattern-Oriented Tensor Decomposition with Bi-LSTM

Xiaoding Guo , Hongli Zhang , Lin Ye, and Shang Li

Harbin Institute of Technology, Harbin, China

Correspondence should be addressed to Hongli Zhang; zhanghongli@hit.edu.cn

Received 12 November 2018; Revised 14 January 2019; Accepted 17 February 2019; Published 7 March 2019

Guest Editor: Haneul Ko

Copyright © 2019 Xiaoding Guo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Online legal consultation plays an increasingly important role in the modern rule-of-law society. This study aims to understand the intention of legal consultation of users with different language expressions and legal knowledge background. A critical issue is a method through which users' legal consultation data are classified and the feature of each category is extracted. Traditional classification methods rely considerably on lexical and syntactic features and frequently require strict sentence formatting, which eliminates substantial energy and may not be universally applicable. We aim to extract the patterns of users' consultation on different categories, which minimally depend on lexical, syntax, and sentence formatting. However, research in this area has rarely been conducted in previous legal advisory service studies. In this study, a classification approach for multiclass users' intention based on pattern-oriented tensor decomposition and Bi-LSTM is proposed, and each user's legal consulting statement is expressed as a tensor. Moreover, we propose a pattern-oriented tensor decomposition method that can obtain a core tensor that approximates the patterns of users' consultation. These patterns can improve the accuracy of classifying users' intention of legal consultation. We use Bi-LSTM to automatically learn and optimize these patterns. Evidently, Bi-LSTM with a pattern-oriented tensor decomposition layer performs better than a recurrent neural network only. Results show that our method is more accurate than the previous work, and the factor matrix and core tensor calculated by the pattern-oriented tensor decomposition are interpretable.

1. Introduction

With the increase in demand for online legal consultation [1], understanding the intention of different users for legal consultation is a problem that must be solved [2]. Different users have various language expressions and levels of legal knowledge [3]; for example, User A inquired as follows: "He sneaked into my house and stole three thousand dollars, how to judge?", and User B asked as follows: "Burglary \$2500, how many sentences and fines should be sentenced?" Users A and B described burglary cases amounting to \$3000. These users expressed the same intention of legal consultation and could be provided with the same category. Thus, a crucial step to understand the users' legal counseling intentions is classifying users' legal consulting statements. Traditional intention classification methods extract sentence features, which rely heavily on lexical and syntactic characteristics,

and generally require sentences to have a strict format. However, users' legal advice data, such as colloquial, disordered, and unprofessional data, are typically disorganized, thereby resulting in numerous difficulties for traditional methods of users' intent classification.

In previous works, understanding users' intent of legal consultation has been rarely accomplished, especially classifying users' intent upon the colloquial, unprofessional, and disordered irregular legal consultation dataset [4]. Figure 1 illustrates the framework of the intent classification model of users' legal consulting statements. Evidently, the problems to be solved mainly include modeling and classifying user legal consulting statements. Traditional intention classification methods are dedicated to feature extraction at the lexical and syntactic layers, and regular datasets with professional knowledge background are typically required to achieve high classification accuracy [5]. Obtaining these datasets

FIGURE 1: Intent classification model for users' legal consulting statements.

requires expert knowledge and consumes substantial human engineering.

Definition 1 (the intention of the users' legal consulting statement). The intention of the users' legal consulting statement is category of consultation involved in it, such as process, assistance, crimes, and judgments on legal cases.

Problem 2. We represent users' legal consulting statement as a tensor; the category of this statement is expressed in the scalar L . Given the dataset of users' legal consulting statements Ω , $\Omega = \{(\chi^{(1)}, L^{(1)}), (\chi^{(2)}, L^{(2)}), \dots, (\chi^{(N)}, L^{(N)})\}$, where $\chi^{(n)}$ represents the n th statement in Ω and $L^{(n)}$ indicates corresponding categories of $\chi^{(n)}$. Train a model $\phi(\chi)$, which is used to classify and predict the category of a new legal consulting statement χ .

We define the intention of users' legal consulting statements as Definition 1. This article formalizes the problem of understanding users' legal intention as Problem 2. This study proposes a new method for understanding users' intention of legal consultation. In terms of modeling methods for user legal consulting statements, we propose a pattern-oriented tensor decomposition method. We focus on extracting the patterns of user legal consultation, rather than features in the lexical and syntactical levels for different categories. These patterns can be regarded as a kind of data structure and are less dependent on vocabulary, grammar, and sentence formatting than traditional intention classification methods. The pattern-oriented tensor decomposition method is used to extract structured information in a users' legal consulting statement, and the structured information approximates the user legal consulting patterns. For example, we denote the user legal consulting statement as the tensor χ and the user consultation pattern derived by Bi-LSTM as ν . Then, we use χ and ν as inputs of the pattern-oriented tensor decomposition

method and obtain a core tensor $\bar{\chi}$, which is construed as the structured information of tensor χ and approximated to pattern ν . $\bar{\chi}$ carries not only the vocabulary and syntax data but also the structured information of χ .

In terms of classification model optimization, this study proposes a user legal consulting intent classification method on the basis of Bi-LSTM and pattern-oriented tensor decomposition. We use Bi-LSTM to automatically learn and optimize users' legal consultation patterns and obtain patterns that are highly favorable for classifying users' legal consultation intention. Moreover, Bi-LSTM, which passes the pattern-oriented tensor decomposition layer, is more accurate in classifying users' consultation intent and more relaxed on the datasets than directly using the users' legal consulting statement tensor as the input to Bi-LSTM. Furthermore, the core tensor obtained by the pattern-oriented tensor decomposition method contains structured information that approximates the legal consultation patterns of different categories. Simultaneously, the core tensor dimension is considerably lower than the original one. The core tensor can be regarded as the main structured information of the original tensor for user intention classification.

The main contributions of this study are summarized as follows:

- (i) *This study proposes a new feature extraction model for users' legal consulting data.* In contrast to the traditional feature extraction method based on the vocabulary and syntax, this study proposes a new pattern-oriented tensor decomposition method, which extracts the core tensor that represents the main structured information of users' legal consultation data from the original tensor. The core tensor is approximate to users' consultation patterns of different categories. The consultation patterns

are beneficial to classifying users' legal consultation intentions. In particular, in comparison with the original tensor, the core tensor obtained by the pattern-oriented tensor decomposition method can improve the accuracy of the Bi-LSTM classification algorithm.

- (ii) *This study proposes a new intent classification approach that combines Bi-LSTM with pattern-oriented tensor decomposition method.* In this study, we use the pattern-oriented tensor decomposition layer to extract core tensors from the original ones in accordance with the user consultation patterns for each category. We use Bi-LSTM to autonomously learn and optimize these patterns. Then, we obtain legal consultation patterns and core tensors, which are beneficial to classifying users' legal consultation intention. Our proposed method depends less on lexical, syntactic, and sentence formatting of datasets and is more universal than traditional intent classification schemes that rely heavily on datasets.
- (iii) *This study proposes a new optimization method for pattern-oriented tensors on the basis of Bi-LSTM.* We derive the partial derivative of the error function for pattern tensors in Bi-LSTM in accordance with pattern-oriented tensor decomposition method and propagation equations of Bi-LSTM. Furthermore, we use the hyperparameter optimization strategy in Bi-LSTM to continuously update the value of pattern tensors. This approach is conducted to guide various training sample core tensors in improving the classification accuracy of the model.

This paper is organized as follows: Section 2 mainly describes the research procedure on classifying texts in the legal field in recent years and related works on intent recognition. Section 3 mainly introduces related background knowledge, such as several relevant methods, definitions, and notations. Section 4 details the method proposed in this study for user legal advice intent, that is, Bi-LSTM with pattern-oriented tensor decomposition method. Section 5 presents the relevant comparison experiments and result analysis.

2. Related Work

In recent years, research on the understanding of user intent based on deep neural networks and tensor layers in the field of legal services has been rarely conducted. In the past ten years, researchers have concentrated on investigating the classification of legal related documents in the field of law and computer intersection [6]. Text classification in the legal field includes classification and understanding of legal cases, judgment documents, entities involved in legal cases, and laws and regulations.

In [7], Sulea, Zampieri, and Malmasi studied the application of text classification in the legal field for professionals. The authors proposed a method for predicting the judgment of the Supreme Court in France on the basis of machine learning algorithms and statistics and suggested an accurate case-like case retrieval technique and weight fluctuation

algorithm for case influence over time. The SVM algorithm was mainly used to complete the classification of relevant documents in legal cases, and the judgment for the legal cases was realized. In [8], Sarwar, Karim, and Naem studied the software copyright dispute between a user and a software program owner through a semisupervised machine learning algorithm; in addition, the authors predicted and judged the software copyright dispute that may be violated by the software after the user obtained the software license. Copyright disputes in using software licenses are common problems. After users obtain software licenses, they can use the software for a period in accordance with software usage rules.

In [9], Galgani and Hoffmann proposed a method for classifying legal references through incremental knowledge acquisition. This method can be automated to extract the main objectives from the legal text summary. These authors created considerable training and test corpora for legal citation classification in the legal field of Australian court judgment report, which is considered of high quality under Australian law. A specialized legal knowledge base in the field, which uses machine learning algorithms, is utilized to classify legal references. In [10], Xiong studied the automatic classification system in the field of Chinese legal texts. For Chinese legal documents, traditional Chinese character documents cannot be used to model Chinese legal documents. Otherwise, dimensional explosion and computational complexity will heighten. Xiong proposed a legal document clustering method on the basis of latent semantic analysis to diminish the dimension of legal text features. In addition, Xiong established a Chinese taxonomic automatic classification system in accordance with the second dimensionality reduction method based on the foundation of latent semantic analysis.

In [11], Maat, Krabben, and Winkels used machine learning algorithms to classify sentences in the Dutch legal library and compared the results of the classification with the legal sentence classification outcomes on the basis of traditional pattern classifiers. The legal sentence classifier based on machine learning algorithm has higher accuracy than the pattern-based classifier given the accurate modeling of legal sentences and feature extraction. In [12], Bartolini proposed a management labeling system for Italian law. The method aims to cluster the full text by representing redundant long documents in the vector form and achieve document classification. It uses the treaty and article as clustering units and presents clustering experiment results in a tree diagram form.

In the general text classification field, researchers have conducted substantial research [13]. Traditional machine learning algorithms and deep neural networks are used in text classification. From the perspective of machine learning, Nigam, McCallum, and Thrun improved the accuracy of learning text classifiers by using considerable unlabeled documents to augment few marked documents [14]. This method is necessary because obtaining text labels for text classification is costly in practice. However, considerable unlabeled documents are particularly easy to obtain. Their article uses an EM-based approach to learn and mark unlabeled

documents. The algorithm first uses a Bayesian classifier to probabilistically mark unlabeled documents, followed by a small amount. Subsequently, the system counts the expected values of the tagged document, creates a tag classifier on the basis of all documents, and iterates until it converges. From the perspective of deep neural networks, Kim proposed TextCNN that is based on convolutional neural networks for text classification and prediction [15]. Donahue proposed a structure based on recurrent neural networks for text classification, that is, TextRNN [16]. Dzmitry proposed an attention structure for deep neural networks. Attention layer discovers the association between input and output by adding weight parameters [17].

3. Preliminaries

In this section, we introduce several related methods, definitions, and notations. Section 3.1 presents the basic definitions and notations involved in this study. Section 3.2 provides a detailed explanation of the tensor decomposition operation.

3.1. Definitions and Notations. In this study, we present user legal consulting statements in tensors. The pattern-oriented tensor decomposition method is used to decompose these tensors, and the obtained core tensors are used in the subsequent deep neural network classification model. A tensor is a data structure similar to a vector or matrix [18, 19]. Tensor decomposition is a dimensionality reduction operation on the tensor [20, 21]. Similar to principal component analysis and singular value decomposition methods, tensor decomposition methods are devoted to extracting the main structure and compositional information in the original tensor [22, 23].

The tensor is actually a multidimensional array [24], and we use the Euler script letters (χ) to represent the tensor. We refer to the tensor dimension and number of tensor dimension as modes and order, respectively [22]. Scalar, vector, and matrix are denoted in lowercase (a), bold lowercase (**a**), and uppercase letters (A), correspondingly; the transposition of matrix A in A^T ; and unit matrices in I . A square matrix with elements of 1 is represented by $\mathbf{1}$.

Definition 3 (outer product). The outer product of two vectors $\mathbf{a} \in \mathbb{R}^I$ and $\mathbf{b} \in \mathbb{R}^J$ is expressed as $A = \mathbf{a} \circ \mathbf{b}$, $A \in \mathbb{R}^{I \times J}$, that is,

$$A(i, j) = \mathbf{a}(i) \mathbf{b}(j). \quad (1)$$

Definition 4 (Kronecker product). The Kronecker product of two matrices $P \in \mathbb{R}^{I \times J}$ and $Q \in \mathbb{R}^{K \times L}$ is denoted as $P \otimes Q$, which is an $IK \otimes JL$ matrix.

$$P \otimes Q = \begin{bmatrix} P(1,1)Q & P(1,2)Q & \cdots & P(1,J)Q \\ P(2,1)Q & P(2,2)Q & \cdots & P(2,J)Q \\ \vdots & \vdots & \ddots & \vdots \\ P(I,1)Q & P(I,2)Q & \cdots & P(I,J)Q \end{bmatrix} \quad (2)$$

Definition 5 (Khatri-Rao product). The Khatri-Rao product of $P \in \mathbb{R}^{I \times R}$ and $Q \in \mathbb{R}^{J \times R}$ is denoted as $P \odot Q$, which is an $IJ \times R$ matrix, that is,

$$P \odot Q = [P(:, 1) \otimes Q(:, 1) \cdots P(:, r) \otimes Q(:, r) \cdots P(:, R) \otimes Q(:, R)]. \quad (3)$$

Definition 6 (N -mode product). Given N -mode tensor $\chi \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and matrix $A \in \mathbb{R}^{I_n \times J}$, the n -mode product is denoted as $\kappa = \chi \times_n A$, $\kappa \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N}$, that is,

$$\begin{aligned} & \kappa(i_1, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N) \\ &= \sum_{m=1}^{I_n} \chi(i_1, \dots, i_{n-1}, m, i_{n+1}, \dots, i_N) A(m, j). \end{aligned} \quad (4)$$

Definition 7 (N -mode matricization). Given an N -mode tensor $\chi \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, the n -mode matricization of χ is denoted as $\chi_n \in \mathbb{R}^{I_n \times I_1 \cdots I_{n-1} I_{n+1} \cdots I_N}$. The calculation method aims to fix the n th mode and form the elements of other modes into a long matrix.

Definition 8 (Frobenius norm of a tensor). Given an N -mode tensor $\chi \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, the Frobenius norm of χ is denoted as $\|\chi\|_F$, that is,

$$\|\chi\|_F = \sqrt{\sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} \chi(i_1, \dots, i_N)^2}. \quad (5)$$

3.2. Tensor Decomposition. Tensor decomposition is a process of approximating a tensor into a core tensor and several factor matrices [25]. In Figure 2, given an N -mode tensor $\chi \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, the formal expression of tensor decomposition on χ is

$$\chi \approx \zeta \times_1 U_1 \times_2 U_2 \times \cdots \times_N U_N \quad (6)$$

where $\{U_n\}$ is a set of factor matrices, $U_n \in \mathbb{R}^{I_n \times I_n}$. The factor matrices are all column orthogonal ones [19]. Furthermore, ζ is the core tensor, $\zeta \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. The tensor decomposition methods minimize the objective function φ [26], where

$$\varphi = \left\| \chi - \zeta \prod_n U_n \right\|_F. \quad (7)$$

4. Our Approach

This paper proposes Bi-LSTM with pattern-oriented tensor decomposition method for intention classification of users' legal consulting statements. In Section 4.1, the pattern-oriented tensor decomposition method extracts the core tensor $\tilde{\chi}$ from the original tensor χ under the guidance of the pattern tensor ν , in order to make $\tilde{\chi}$ approximate ν . In Section 4.2, Bi-LSTM continually optimizes pattern tensor ν so that $\tilde{\chi}$ carries a specific tensor structured and elemental information in χ . This information is most conducive to

FIGURE 2: Framework of tensor decomposition.

FIGURE 3: Bi-LSTM with pattern-oriented tensor decomposition method.

improving the accuracy of the intent classification model of users' legal consulting statements.

As shown in Figure 3, Bi-LSTM controls the process of pattern-oriented tensor decomposition by optimizing pattern tensor ν . Bi-LSTM continually optimizes pattern tensor ν , while core tensor $\tilde{\chi}$ continues to approach ν through the pattern-oriented tensor decomposition method. Finally, ν becomes the pattern tensor that can make the Bi-LSTM model reach high accuracy, and $\tilde{\chi}$ is the core tensor that is beneficial for improving the accuracy of the subsequent classification model under the guidance of tensor pattern ν .

4.1. Pattern-Oriented Tensor Decomposition Method. The pattern-oriented tensor decomposition method decomposes tensor χ into core tensor $\tilde{\chi}$ and factor matrices $\{U_k\}$ and

$\{C_k\}$, thus making the core tensor $\tilde{\chi}$ approach the users' legal consultation pattern ν , that is, the core tensor $\tilde{\chi}$ and the users' legal consultation pattern ν demonstrate a similar tensor structure. The subsequent Bi-LSTM classification model controls pattern-oriented tensor decomposition by continuously optimizing the pattern tensor ν . This situation implies that the core tensor $\tilde{\chi}$ is more advantageous than users' legal consultation data tensor χ in terms of enhancing the accuracy in classifying users' legal consultation intention. Simultaneously, $\tilde{\chi}$ achieves the dimension reduction effect, which considerably reduces the calculation time and space.

The framework of the pattern-oriented tensor decomposition method is depicted in Figure 4. In this study, the problems to be solved by the pattern-oriented tensor decomposition method are defined as Problems 9 and 10.

FIGURE 4: Framework of pattern-oriented tensor decomposition method.

Problem 9. Given a tensor $\chi \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$ and a pattern tensor $\nu \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_K}$, find two factor matrix sets $\{C_k\}$ and $\{U_k\}$, $C_k \in \mathbb{R}^{I_k \times H_k}$, $U_k \in \mathbb{R}^{J_k \times H_k}$, where $\{C_k\}$ and $\{U_k\}$ satisfy Conditions 1 and 2 simultaneously.

Condition 1. The factor matrix sets $\{C_k\}$ and $\{U_k\}$ minimize the following target function.

$$\phi = \left\| \chi \prod_k^K \times_k C_k - \nu \prod_k^K \times_k U_k \right\|_F \quad (8)$$

Condition 2. Factor matrices in $\{C_k\}$ and $\{U_k\}$ are orthogonal matrices, that is,

$$\begin{aligned} C_k C_k^T &= I \\ C_k^T C_k &= I \\ \text{and } U_k U_k^T &= I \\ U_k^T U_k &= I. \end{aligned} \quad (9)$$

Problem 10. Given tensor $\chi \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$ and two factor matrix sets $\{C_k\}$ and $\{U_k\}$, $C_k \in \mathbb{R}^{I_k \times H_k}$, $U_k \in \mathbb{R}^{J_k \times H_k}$, where $\{C_k\}$ and $\{U_k\}$ satisfy Conditions 1 and 2 simultaneously, find a core tensor $\tilde{\chi}$, $\tilde{\chi} \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_K}$ that minimizes the target function ψ , where

$$\psi = \left\| \chi \prod_k^K \times_k C_k - \tilde{\chi} \prod_k^K \times_k U_k \right\|_F. \quad (10)$$

In Problem 10, we can calculate the value of $\tilde{\chi}$ by setting the partial differential of function ψ with respect to $\tilde{\chi}$ to 0. The specific conclusion is presented in Theorem 12. In Appendix A, Proof A.0.1 provides the proof of Lemma 11, and Proof A.0.2 provides the solution of (11) in Theorem 12.

Lemma 11. Given that the Frobenius function $\eta = \|(\tilde{\chi} \prod_k^K \times_k U_k) \prod_k^K \times_k U_k^T \tilde{\chi}^T\|_F^2$, where $\tilde{\chi} \in \mathbb{R}^{J_1 \times J_2 \times \dots \times J_K}$, $U_k \in \mathbb{R}^{J_k \times H_k}$, and U_k satisfies (9), the partial differential of function η to $\tilde{\chi}$ is $\varepsilon(\tilde{\chi} \prod_k^K \times_k U_k) \prod_k^K \times_k U_k^T$, where ε is a constant.

Theorem 12. Given $\{C_k\}$ and $\{U_k\}$ in Problem 9, we can obtain the optimal solution of $\tilde{\chi}$ that minimizes the target function ψ in Problem 10.

$$\tilde{\chi} = \left(\chi \prod_k^K \times_k C_k \right) \prod_k^K \times_k U_k^T \quad (11)$$

The following part is the process of calculating the sets of factor matrices $\{C_k\}$ and $\{U_k\}$ in Problem 9. Under the constraint of Conditions 1 and 2, we can calculate the optimal solution of function ϕ by using alternating least squares (ALS), Lemma 13, Proof B.0.3, and Proof B.0.4. In Appendix B, Proof B.0.3 elaborates the proof process of Lemma 13, and Proof B.0.4 gives the solution of Problem 9.

Lemma 13. Given the function $\delta = \text{tr}((\chi \prod_k^K \times_k C_k)^T (\chi \prod_k^K \times_k C_k))$, where $\chi \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$, $C_k \in \mathbb{R}^{I_k \times H_k}$, and C_k satisfies (9),

the partial differential of function δ to C_{k_0} , where $k_0 \in [1, K]$, is $\varepsilon(\chi \prod_{k \neq k_0}^K \times_k C_k)^T (\chi \prod_k^K \times_k C_k)$, where ε is a constant.

The ALS algorithm aims to use the partial derivative of the remaining variable while fixing other variables and find the value of the variable when the partial derivative is zero. Then, the value is substituted for the original objective function. Similarly, the values of other variables are calculated using the same process. The ALS continuously iterates until the calculation error is tolerable. The process of calculating the optimal solution $\{C_k\}$ and $\{U_k\}$ that minimize target function (8) under constraint Conditions 1 and 2 is demonstrated in Proof B.0.4.

Algorithm 1 demonstrates the process of the pattern-oriented tensor decomposition method. The present study uses the ALS algorithm to optimize the parameters involved in Algorithm 1. The input of Algorithm 1 is the tensor χ that represents users' legal consulting statement and the user legal consultation pattern ν , which is beneficial to classifying users' legal consultation intention. The outputs of Algorithm 1 are the core tensor $\tilde{\chi}$ and the corresponding factor matrices $\{C_k\}$ and $\{U_k\}$. In addition, $\tilde{\chi}$ can be interpreted as a feature map of the original tensor χ in the space determined by core tensor ν . That is, the original users' legal consulting statement is mapped to the feature space that is beneficial for classifying users' legal consultation intention. Then, we can accurately understand users' legal consultation intention.

In Line 1 of Algorithm 1, we initialize the sets of factor matrices related to the core tensor $\tilde{\chi}$. Then we use the ALS algorithm to calculate the optimal solution of $\{C_k\}$ and $\{U_k\}$ that minimize the value of (8) under Condition 2. Furthermore, *thresholdvalue* in Line 2 represents the value of number of iterations we set for the ALS algorithm. Function *transfun* in Line 4 embodies the calculation process of (B.3) in Proof B.0.4 of Appendix B. Line 5 completes the SVD decomposition of the transition variable G_{k_0} , which corresponds to (B.4) in Proof B.0.4 of Appendix B. Moreover, Line 6 presents the method for calculating C_{k_0} that minimizes (8) while fixing $\{C_k\}$ where $k \neq k_0$ and $\{U_k\}$. Similarly, Line 8 to Line 11 demonstrate the process of calculating U_{k_0} to minimize (8) while fixing $\{U_k\}$ where $k \neq k_0$ and $\{C_k\}$. In Line 14, function *calculate* represents the calculation of $\tilde{\chi}$ which is the core tensor of users' legal consulting statement χ . $\tilde{\chi}$ is interpreted as a result of the tensor decomposition of χ directed to users' legal consultation pattern ν .

4.2. Optimization Method of Users' Legal Consulting Pattern.

In this study, we use the Bi-LSTM [27] to optimize users' legal consultation pattern ν and ensure that the final calculated ν is a favorable users' legal consultation pattern for the classification model of users' legal consulting intention. Notably, the optimization function of Bi-LSTM is Rmsprop. The following section presents the training process of Bi-LSTM using Rmsprop as its optimization function:

- (i) We use the initial user legal consultation pattern ν and the core tensor set $\{\tilde{\chi}^{(n)}\}$ which represents user's legal consulting statements as the input of Bi-LSTM. Each

core tensor $\tilde{\chi}^{(n)}$ in the core tensor set $\{\tilde{\chi}^{(n)}\}$ is the result of the pattern-oriented tensor decomposition method while using the corresponding original tensor $\chi^{(n)}$ and the user legal consultation pattern ν as the input. $\tilde{\chi}^{(n)}$ approaches the user legal consultation pattern ν on the layer of tensor structure.

- (ii) In this study, the output $\{h_d^{(n)}\}$ of Bi-LSTM is used as the input of the softmax layer to realize the mapping of output vectors to categories $\{L^{(n)}\}$ of users' legal consulting statement $\{\chi^{(n)}\}$, and d is the number of hidden layers. Moreover, the cross entropy is used as the loss function *LossF* for calculating the error.
- (iii) By propagating the forward and reverse between LSTM units, using the formulas of the error back-propagation in Bi-LSTM over time, and the error inverse propagation between the hidden layers of Bi-LSTM, we calculate the partial derivative of the loss function *LossF* with respect to the weight matrix $\{w_d\}$, the bias term $\{b_d\}$, and the users' legal consulting pattern ν , that is, $\partial \text{LossF} / \partial w_d$, $\partial \text{LossF} / \partial b_d$, and $\partial \text{LossF} / \partial \nu$, correspondingly.
- (iv) The Rmsprop optimization function is used to continuously optimize and iterate the abovementioned parameters $\{w_d\}, \{b_d\}$, and ν using the value of $\partial \text{LossF} / \partial w_d$, $\partial \text{LossF} / \partial b_d$, and $\partial \text{LossF} / \partial \nu$, correspondingly. Finally we obtain the value of the weight matrix $\{w_d\}$, bias term $\{b_d\}$, and users' legal consultation pattern ν . These parameters are favorable for users' legal consultation intention classification model based on Bi-LSTM.

4.2.1. Method for Calculating the Partial Derivative of *LossF* to ν . This study proposes a method for solving the partial derivative of the loss function *lossF* with respect to users' legal counseling pattern ν . Directly calculating the partial derivative of *lossF* to ν is difficult. However, we can indirectly determine the partial derivative of the loss function *lossF* on users' legal consultation pattern ν by using the total and indirect derivative rules.

In this study, we use tensor χ which represents users' legal consulting statement, the factor matrices $\{C_k\}$ and $\{U_k\}$, and the core tensor $\tilde{\chi}$ which approaches users' legal consultation pattern ν on the layer of tensor structure as transition variables. $\tilde{\chi}$, $\{C_k\}$, and $\{U_k\}$ are calculated through the pattern-oriented tensor decomposition method with χ and ν as its inputs. The transition variables χ , $\tilde{\chi}$, $\{C_k\}$, and $\{U_k\}$ transform the derivative problem $\partial \text{lossF} / \partial \nu$ into the Sylvester problem, we use the Hessenberg-Schur algorithm to solve the Sylvester matrix equation, and finally the partial derivative of loss function *lossF* with respect to users' legal consulting pattern ν is obtained.

Lemma 14. Given tensors $\{\chi^{(n)}\}$ which represent users' legal consulting statements, users' legal consultation pattern ν , and the corresponding factor matrices $\{C_k^{(n)}\}$, $\{U_k^{(n)}\}$ and the core tensors $\{\tilde{\chi}^{(n)}\}$ which are obtained through the pattern-oriented

Input: Tensor χ which represents users' legal consulting statement, $\chi \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$, and tensor ν which is users' legal consultation pattern, $\nu \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$.

Output: The core tensor $\tilde{\chi}$ which is close to users' legal consultation pattern ν in the layer of the tensor structure, $\tilde{\chi} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$, and its corresponding factor matrices $\{C_k\}$ and $\{U_k\}$, $C_k \in \mathbb{R}^{I_k \times H_k}$, $H_k \in \mathbb{R}^{I_k \times H_k}$.

```

1 Initialize factor matrices  $\{C_k\}$  and  $\{U_k\}$ ;
2 for  $i = 1$  to thresholdvalue do
3   for  $k_0 = 1$  to  $K$  do
4      $G_{k_0} = \text{transfun}(\chi, \nu, \{C_k\}, \{U_k\}, k_0)$ ;
5      $P_{k_0}, S_{k_0}, Q_{k_0} = \text{SVD}(G_{k_0})$ ;
6      $C_{k_0} = P_{k_0} Q_{k_0}^T$ ;
7   end
8   for  $k_0 = 1$  to  $K$  do
9      $G_{k_0} = \text{transfun}(\chi, \nu, \{U_k\}, \{C_k\}, k_0)$ ;
10     $P_{k_0}, S_{k_0}, Q_{k_0} = \text{SVD}(G_{k_0})$ ;
11     $U_{k_0} = P_{k_0} Q_{k_0}^T$ ;
12  end
13 end
14  $\tilde{\chi} = \text{calculate}(\chi, \{C_k\}, \{U_k\})$ ;
15 return  $\tilde{\chi}, \{C_k\}, \{U_k\}$ ;

```

ALGORITHM 1: Pattern-oriented tensor decomposition method.

tensor decomposition method with $\{\chi^{(n)}\}$ and ν as its inputs, the partial derivative of loss function lossF with respect to the users' legal consulting pattern ν is obtained using

$$\frac{\partial \text{lossF}}{\partial \nu} = \sum_n \left(\text{lossF}_{\tilde{\chi}^{(n)}} * \text{calx}_{\nu}^{(n)} \right) \quad (12)$$

where $\text{lossF}_{\tilde{\chi}^{(n)}} = \partial \text{lossF} / \partial \tilde{\chi}^{(n)}$ and $\text{calx}_{\nu}^{(n)} = \sum_k^K \text{funC}(C_k^{(n)})_{\nu} + \sum_k^K \text{funU}(U_k^{(n)T})_{\nu}$. Functions $\text{funC}(C_k^{(n)})_{\nu}$ and $\text{funU}(U_k^{(n)T})_{\nu}$ meet the following limitations.

$$\text{funC}(C_k^{(n)})_{\nu} = \chi^{(n)} \quad (13)$$

$$\times_k \frac{\partial C_k^{(n)}}{\partial \nu} \prod_{k_1 \neq k}^K \times_{k_1} C_{k_1}^{(n)} \prod_{k_2}^K \times_{k_2} U_{k_2}^{(n)T}$$

$$\text{funU}(U_k^{(n)T})_{\nu} = \chi^{(n)} \prod_{k_2}^K \times_{k_2} C_{k_2}^{(n)} \quad (14)$$

$$\times_k \frac{\partial U_k^{(n)T}}{\partial \nu} \prod_{k_1 \neq k}^K \times_{k_1} U_{k_1}^{(n)T}$$

Proof C.0.5 in Appendix C elucidates the proof of Lemma 14. Using (B.2), (B.3), and (B.4) in Proof B.0.4 of Appendix B, we can obtain $\partial G_k^{(n)} / \partial \nu = (\chi^{(n)} \prod_{k_1 \neq k}^K \times_{k_1} C_{k_1}^{(n)})_k (\mathfrak{S} \prod_{k_2}^K \times_{k_2} U_{k_2}^{(n)T})_k$ and $G_k^{(n)} = C_k^{(n)} G_k^{(n)T} C_k^{(n)}$, $\mathfrak{S}$ is the identity tensor, $\mathfrak{S} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$. The next part is the method for solving $\partial C_k^{(n)} / \partial \nu$. $\partial U_k^{(n)T} / \partial \nu$ and $\partial C_k^{(n)} / \partial \nu$ are calculated similarly. We determine the partial derivative of ν on both sides of the abovementioned equation and obtain that $\partial G_k^{(n)} / \partial \nu - C_k^{(n)} (\partial G_k^{(n)T} / \partial \nu) C_k^{(n)} = (\partial C_k^{(n)} / \partial \nu) G_k^{(n)T} C_k^{(n)} +$

$C_k^{(n)T} G_k^{(n)} (\partial C_k^{(n)} / \partial \nu)$, which is the classic Sylvester equation and can be calculated using the Hessenberg-Schur algorithm.

Algorithm 2 demonstrates the optimization method for users' legal consulting pattern in this study. In Line 2, *max_iter* represents the training times of the Bi-LSTM used. Lines 4 to 12 are the training steps for the Bi-LSTM model. In Line 6, *batch_size* represents the number of samples per small batch training. Function *TPTD* in Line 7 denotes the pattern-oriented tensor decomposition method. Line 8 presents the forward propagation process in Bi-LSTM on forward and backward layers. Line 9 elucidates the backpropagation process of errors over time and neural network layers in Bi-LSTM. We use Rmsprop in Line 11 as an optimization function to optimize the parameters in Bi-LSTM.

4.2.2. Loss Function lossF and Softmax Layer. In Algorithm 2, we use function *softmax* to calculate the probability that $\chi^{(n)}$ belongs to each category. For intent classification of users' legal consulting statements, the function above is defined as follows.

Definition 15. Given a set of users' legal consulting statement samples and their corresponding outputs of Bi-LSTM $\{(\chi^{(n)}, Y_1^{(n)})\}$, the probability that $\chi^{(n)}$ belongs to each category is calculated using

$$L_{1q}^{(n)} = \text{softmax}(Y_1^{(n)}) = \frac{e^{Y_{1q}^{(n)}}}{\sum_r^Q e^{Y_{1r}^{(n)}}} \quad (15)$$

where $L_{1q}^{(n)}$ represents the q th element of $L_1^{(n)}$.

In this study, the cross entropy is used as the loss function *lossF* to calculate the error of Bi-LSTM. We define *lossF* as follows.

Input: $\{(\chi^{(n)}, L^{(n)})\}$, where $\chi^{(n)}$ is the user's legal consulting statement and $L^{(n)}$ represents category that corresponds to $\chi^{(n)}$, and the size of ν , $\{w_f\}$, $\{w_i\}$, $\{w_o\}$, $\{w_c\}$, $\{b_f\}$, $\{b_i\}$, $\{b_o\}$ and $\{b_c\}$, where $w_f = [w_{fh}, w_{fx}]$, $w_i = [w_{ih}, w_{ix}]$, $w_o = [w_{oh}, w_{ox}]$ and $w_c = [w_{ch}, w_{cx}]$

Output: The optimal users' legal consultation pattern ν , parameters of Bi-LSTM $\{w_f\}$, $\{w_i\}$, $\{w_o\}$, $\{w_c\}$, $\{b_f\}$, $\{b_i\}$, $\{b_o\}$ and $\{b_c\}$.

- 1 Initialize users' legal consultation pattern ν , parameters of Bi-LSTM $\{w_f\}$, $\{w_i\}$, $\{w_o\}$, $\{w_c\}$, $\{b_f\}$, $\{b_i\}$, $\{b_o\}$ and $\{b_c\}$;
- 2 **for** $iter = 1$ to max_iter **do**
- 3 Set $batch_indices$ to zero;
- 4 **while** $batch_indices \leq dataset_size$ **do**
- 5 Set $\partial lossF / \partial \nu$, $\partial lossF / \partial w_f$, $\partial lossF / \partial w_i$, $\partial lossF / \partial w_o$, $\partial lossF / \partial w_c$, $\partial lossF / \partial b_f$, $\partial lossF / \partial b_i$, $\partial lossF / \partial b_o$, $\partial lossF / \partial b_c$ to zero;
- 6 **for** $i = batch_indices$ to $(batch_indices + batch_size)$ **do**
- 7 $\tilde{\chi}^{(i)}, \{C_k\}, \{U_k\} = TPTD(\chi^{(i)}, \nu)$;
- 8 $L_1^{(i)} = Bi_LSTM_forwardprop(\tilde{\chi}^{(i)}, \{w_f\}, \{w_i\}, \{w_o\}, \{w_c\}, \{b_f\}, \{b_i\}, \{b_o\}, \{b_c\})$;
- 9 $\frac{\partial lossF}{\partial \nu}, \frac{\partial lossF}{\partial w_f}, \frac{\partial lossF}{\partial w_i}, \frac{\partial lossF}{\partial w_o}, \frac{\partial lossF}{\partial w_c}, \frac{\partial lossF}{\partial b_f}, \frac{\partial lossF}{\partial b_i}, \frac{\partial lossF}{\partial b_o}, \frac{\partial lossF}{\partial b_c} +=$
 $Bi_LSTM_backprop(\tilde{\chi}^{(i)}, L_1^{(i)}, L^{(i)}, \nu, \{w_f\}, \{w_i\}, \{w_o\}, \{w_c\}, \{b_f\}, \{b_i\}, \{b_o\}, \{b_c\})$;
- 10 **end**
- 11 $\nu, \{w_f\}, \{w_i\}, \{w_o\}, \{w_c\}, \{b_f\}, \{b_i\}, \{b_o\}, \{b_c\} = Rmsprop(\nu, \{w_f\}, \{w_i\}, \{w_o\}, \{w_c\}, \{b_f\}, \{b_i\}, \{b_o\}, \{b_c\},$
 $\frac{\partial lossF}{\partial \nu}, \frac{\partial lossF}{\partial w_f}, \frac{\partial lossF}{\partial w_i}, \frac{\partial lossF}{\partial w_o}, \frac{\partial lossF}{\partial w_c}, \frac{\partial lossF}{\partial b_f}, \frac{\partial lossF}{\partial b_i}, \frac{\partial lossF}{\partial b_o}, \frac{\partial lossF}{\partial b_c})$;
- 12 **end**
- 13 **end**
- 14 **return** $\nu, \{w_f\}, \{w_i\}, \{w_o\}, \{w_c\}, \{b_f\}, \{b_i\}, \{b_o\}$ and $\{b_c\}$;

ALGORITHM 2: Optimization method of users' legal consulting pattern.

Definition 16. Given a set of users' legal consulting statement samples and their corresponding categories $\{(\chi^{(n)}, L^{(n)})\}$. The estimated category of $\chi^{(n)}$ is $L_1^{(n)}$, which is calculated using Bi-LSTM; then

$$lossF(\{L^{(n)}\}, \{L_1^{(n)}\}) = \sum_n \sum_q L_q^{(n)} \log L_{1q}^{(n)} \quad (16)$$

where N represents the number of samples of users' legal consulting statements. q denotes the dimension of $L^{(n)}$ and $L_1^{(n)}$, that is, the number of categories.

5. Empirical Results

We provide the results of the deep learning model with pattern-oriented tensor decomposition proposed in this study on actual datasets. The experiment verifies that the Bi-LSTM model with pattern-oriented tensor decomposition can accurately classify and understand users' legal consulting sentences and intentions comprehensively. Bi-LSTM with a pattern-oriented tensor decomposition layer is more efficient, interpretable, and instructive than traditional recurrent neural networks.

5.1. Data Description. The data used in this study are mainly online users' legal consulting statements. Our main data sources include the China Legal Business Consulting website and various public legal consulting service platforms at the local level. In this study, approximately 150,000 legal consultations are obtained from all over China from 2008 to 2018.

These data have been manually labeled under the professional legal background and divided into 28 categories, including various common legal disputes, such as divorce and contract disputes, property transfer, and loan compensation.

Moreover, this study conducts a rigorous statistical analysis of the collected datasets and discovers certain interesting data. From 2008 to 2018, the public online legal advice issues are mainly concentrated on labor and personnel, civil, contract, and property disputes; marriage relationship; and creditor's rights debt.

Figure 5(a) displays the distribution of different categories of partial users' legal consulting statements. Evidently, the legal disputes that people aim to solve through online legal consultation have evident tendencies, mainly in the civil aspects, such as marriage and loan disputes and property division. By contrast, major or extraordinarily serious criminal offenses are extremely rare.

5.2. Baseline Approaches. In order to understand the intent of users' legal consulting statements, we proposed Bi-LSTM with pattern-oriented tensor decomposition method. However, in previous studies, research on the understanding of users' legal consulting intention using deep neural networks and tensors has been rarely mentioned. We establish the experimental comparison works of this study on the following basic points:

- (i) From the perspective of deep neural networks, we use the latest neural networks for comparison with Bi-LSTM, including TextCNN [15] and TextCNN attention [17] which are based on convolutional neural

FIGURE 5: Statistics on dataset of users' legal consulting statements.

networks [28], TextRNN [16] and TextRNN attention [17] which are based on recurrent neural networks, and LSTM, GRU [29], Bi-GRU [30].

- (ii) From the perspective of the tensor decomposition layer, we use common tensor decomposition algorithms for comparison with pattern-oriented tensor decomposition method, including Tucker and CP tensor decomposition algorithms.

Through the first point above, we show the performance of Bi-LSTM relative to other deep neural networks on intention classification of users' legal consulting statements. Through the second point above, we show the superiority of pattern-oriented tensor decomposition method compared to other unsupervised tensor decomposition algorithms.

5.3. Feature Extraction. In this study, numerous preprocessing operations are performed on the obtained users' legal consulting statements. The preprocessing operation can be divided into two main steps:

- (i) Module definition of users' legal consulting statement. In this study, each users' legal consulting statement is represented as a three-dimensional tensor. We divide user's legal consultation into five modules, namely, subject, object, motivation, behavior, and consequences of consultation. Each module contains multiple vocabularies and is represented by a matrix of predetermined dimensions. The vertical dimension of the matrix demonstrates vocabularies contained in the module, and the horizontal dimension demonstrates embedding vectors of these vocabularies.
- (ii) Quantitative representation of users' legal consulting statement. This study first performs word segmentation on users' legal consulting statements and remove Chinese punctuation marks, stop words,

redundant vocabularies, and other basic operations because the data collected are all Chinese users' legal consulting statements. Furthermore, this study divides each users' consulting statement in the dataset into five modules in accordance with the previous step. This study represents the vocabularies in users' legal consulting statements in embeddings, that is, word-embedding operation. On this basis, this study instantiates each module of users' legal consulting statement, vectorizes each vocabulary, and represents each user's legal consulting statement in tensor.

The users' legal consulting statement is represented by a three-dimensional tensor. In Figure 6, the first dimension of the tensor represents modules in the statement. The second dimension represents meaningful vocabulary contained in each module. The vocabularies are derived from the original statement through removing redundant, meaningless, and repeated words. The third dimension represents the word embedding corresponding to each word.

We divide each user's legal consulting statement into five modules, namely, subject, object, motivation, behavior, and consequences of consultation. Each module in the users' consulting statement exhibits multiple entity objects. For example, in the users' legal consulting statement: "Xiao Wang repeatedly threatened me with a knife and took me more than 30,000 yuan. What crime should he sue?", the subjects of the consultation module are "me" and "Xiao Wang". Then, the object of the consultation module is "30,000 yuan". The motivation, behavior, and consequence of the counseling module correspond to "crime", "threatened", and "knife".

The word vector generation model is trained under large Chinese corpus. The Chinese Wikipedia and news from multiple websites, such as Tencent, Sohu, and Sogou, are used as corpora for Chinese word vector training [31]. Then we use the word2vec tool proposed by Google to train Chinese word

FIGURE 6: Tensor representation of users' legal consulting statements.

vectors [32]. Word2vec converts one-hot vectors in corpus into low-dimensional dense vectors. The word-embedding operations ensure that users' legal consulting statements can be processed using the Bi-LSTM model with pattern-oriented tensor decomposition presented in this study [33]. We fix users' legal consulting statements to the same length because tensors representing them must have the same dimensions.

The length of the users' legal consulting statement in the database is illustrated in Figure 5(b). Evidently, the number of vocabularies for most users' consulting statements is between 15 and 500, except for a particularly small number of users' legal consulting statements in which the number of vocabularies is higher than 2000 or less than 10. In this study, we set a vocabulary baseline and run users' legal consulting statements with more vocabularies than the baseline. Then, we fill in users' legal consulting statements with fewer vocabularies than the baseline.

5.4. Parameter Adjustment and Experimental Settings. We have implemented a tensor representation of each user's legal consulting statement in the database on the basis of the abovementioned operations. This study uses a TensorFlow development kit to complete the programming of the proposed method. Then, the parameters of the Bi-LSTM model with the pattern-oriented tensor decomposition method proposed in this study are set.

In contrast to the traditional deep neural network algorithms, the important parameters involved in this model include the size of batches while training the neural network and number of layers, neurons in each layer, and iterations of the overall neural network algorithm. Furthermore, these parameters contain users' legal consultation pattern ν . The setting of users' legal consultation pattern ν seriously affects the convergence speed and accuracy of the model. Our experiments show that the classification accuracy of users' legal consulting statement is difficult to increase when the structure of users' legal consultation pattern ν is single; that is, column vectors in ν exhibit a linear relationship.

For all neural networks involved in the experiments of this article, including TextCNN, TextCNN attention, TextRNN, TextRNN attention, LSTM, Bi-LSTM, GRU, and Bi-GRU mentioned in Section 5.2, we trained each of them for

10 epochs with a batch size of 60, a hidden layer size of 512, a hidden layer number of 3, and a learning rate of 0.001. We use the TensorFlow development kit to implement neural networks and use Graphics Processing Unit (GPU) to run programs for a fast computing speed.

5.5. Experimental Results and Analysis. We provide experiments on the baseline in Section 5.2 and our approach for intention classification of users' legal consulting statements in this section. Simultaneously, we provide a detailed explanation of the superiority of Bi-LSTM and the necessity of pattern-oriented tensor decomposition.

Figure 7 indicates that neural networks with tensor decomposition layer converge faster and have higher accuracy than that without it. In fact, this phenomenon is determined by the characteristics of tensor decomposition methods. Tensor decomposition algorithms extract the main structure and element information from the original tensor, while removing redundant information which has strong logical correlation. That is, tensor decomposition weakens the influence of vocabularies with weak relevance to the intention of users' legal consulting statements on the classification model and enhances the influence of strong related vocabularies on it. Moreover, the tensor decomposition layer reduces the dimension of original tensors. This greatly reduces the computational complexity of subsequent deep neural networks. Therefore, the tensor decomposition layer makes neural networks converge faster and achieve higher accuracy.

As can be seen from the pink and cyan curves in Figure 7, Tucker and CP tensor decomposition have basically the same optimization effect on neural networks. This is because CP decomposition is a special case of Tucker decomposition. Tucker tensor decomposition is actually a high-order singular value decomposition (SVD). Tucker decomposition uses the SVD algorithm to iteratively extract the main components of each mode in the original tensor and finally figures out a core tensor and its corresponding factor matrix set. When the core tensor is a diagonal tensor, Tucker decomposition evolves into CP decomposition. Core tensors obtained by Tucker and CP decomposition are weakly interpretable. These methods are all unsupervised tensor decomposition methods. For

(a) Experimental results of algorithms based on LSTM

(b) Experimental results of algorithms based on Bi-LSTM

(c) Experimental results of algorithms based on TextCNN

(d) Experimental results of algorithms based on TextRNN

FIGURE 7: Experiments on Bi-LSTM and other neural networks.

neural network algorithms, Tucker and CP decomposition are not steerable and autonomous learning.

According to the red curves in Figure 7, we can see that the pattern-oriented tensor decomposition layer optimizes neural networks much more than Tucker and CP decomposition. It allows neural network algorithms to converge faster while achieving higher accuracy. The pattern-oriented tensor decomposition algorithm controls the tensor decomposition process through pattern tensors. This algorithm makes the core tensor extracted from the original one approximate the pattern tensor on tensor structure and elements information. On this basis, neural network classification algorithms affect the process of tensor decomposition by continuously optimizing the pattern tensor. These operations ultimately make core tensors carry information that is most conducive

to improving the accuracy of the classification model. Therefore, compared with Tucker and CP tensor decomposition, pattern-oriented tensor decomposition method is more instructive and autonomous learning. Moreover, resulting core tensors are more interpretable. In general, the pattern tensor is a bridge between tensor decomposition and neural networks.

Figure 7(c) demonstrates that TextCNN has lower accuracy than TextRNN, LSTM, and Bi-LSTM in classifying the intention of users' legal consulting statements. This is because the convolution kernel is more concerned with the spatial relationship of input data. Convolutional neural networks only consider the current input while recurrent neural networks consider both the current input and previous inputs. Users' legal consulting statements are sequence data. The

TABLE 1: Accuracy of algorithms based on multiple deep neural networks.

Optimization	Basic algorithm							
	TextCNN	TextCNN attention	TextRNN	TextRNN attention	LSTM	Bi-LSTM	GRU	Bi-GRU
The original method	0.77	0.80	0.79	0.81	0.81	0.83	0.81	0.85
CP TP method	0.82	0.85	0.87	0.86	0.89	0.92	0.90	0.91
Tucker TP method	0.84	0.86	0.85	0.88	0.87	0.90	0.89	0.92
Pattern-oriented TP	0.89	0.91	0.91	0.92	0.93	0.96	0.92	0.94

TABLE 2: F1 score of algorithms based on multiple deep neural networks.

Optimization	Basic algorithm							
	TextCNN	TextCNN attention	TextRNN	TextRNN attention	LSTM	BiLSTM	GRU	Bi-GRU
The original method	0.64	0.70	0.68	0.71	0.70	0.74	0.71	0.77
CP TP method	0.72	0.77	0.80	0.78	0.83	0.87	0.83	0.84
Tucker TP method	0.75	0.77	0.76	0.81	0.80	0.84	0.82	0.86
Pattern-oriented TP	0.83	0.85	0.84	0.86	0.88	0.92	0.86	0.89

meaning of one word is related to words both before and after it. Compared with convolutional neural networks, recurrent neural networks can capture the lexical sequence relationships such as transition, emphasis, and juxtaposition included in statements. Therefore, recurrent neural networks provide a more comprehensive analysis of users' legal consulting statements and achieve higher accuracy.

As can be seen from Figures 7(a), 7(b), and 7(d), LSTM and Bi-LSTM can achieve higher accuracy than TextRNN. Recurrent neural networks are difficult to handle long-distance dependencies. When the input users' legal consulting statement is long, recurrent neural networks may experience gradient disappearance or explosion. LSTM-based neural networks solve the above problem by adding new cell states and gating mechanisms. Bi-LSTM comprehensively considers outputs of the forward and backward LSTM units. Compared with unidirectional LSTM, Bi-LSTM can achieve higher accuracy.

Tables 1 and 2 provide the accuracy and Micro-F1 score of a variety of neural networks for intention classification of users' legal consulting statements. TP stands for tensor decomposition. It can be seen that Bi-LSTM with pattern-oriented tensor decomposition layer has the highest accuracy compared to other algorithms. From the perspective of sequence coding, attention layer can break the limit of fixed-length inputs and calculate the relationship between input sequences and output sequences. Although attention layer adds a series of weight parameters and learns the weight of each element from inputs and outputs sequences, it does not change the structure inside original neural networks. For the problem of intention classification of users' legal consulting statements, attention layer is difficult to compensate for the missing sequence information of TextCNN and the gradient disappearance or explosion problems of TextRNN.

Tables 1 and 2 demonstrate that the pattern-oriented tensor decomposition layer has a greater optimization effect

on LSTM and Bi-LSTM than GRU and Bi-GRU. GRU is a simplification of LSTM. LSTM controls outputs of neural units through the output gate, while GRU passes outputs directly to next neural units. Therefore, GRU converges faster than LSTM. For the optimization of pattern tensors, LSTM is better than GRU. The main reason is that GRU has higher integration and fewer adjustable parameters than LSTM. That is to say, GRU has a relatively limited optimization of pattern tensors.

6. Conclusion

In this study, we propose a new method (i.e., Bi-LSTM with pattern-oriented tensor decomposition) to solve the problem of users' legal intention understanding in the field of legal services. Our method combines deep neural network with tensor decomposition method to complete the classification and deep understanding of users' legal consulting statements. Our method is more instructive and interpretable than the traditional deep neural networks. We propose a new tensor decomposition method that is driven by users' legal consultation patterns and continuously guide the training and update process of deep neural networks.

Experiments show that our technique demonstrates faster convergence and higher accuracy than the traditional deep learning methods by applying the tensor decomposition layer. We will extend our algorithm in future work and increase the dataset to enrich the model parameters and make these parameters universal.

Appendix

A.

We provide the proof of Lemma 11 in Proof A.0.1, and the proof of Theorem 12 in Proof A.0.2. Through these proofs,

Theorem 12 and Lemma 11 proved to be able to solve Problem 10.

Proof A.0.1. In the target function $\eta = \|(\tilde{\chi}\prod_k^K \times_k U_k)\prod_k^K \times_k U_k^T \tilde{\chi}^T\|_F^2$, we use μ to represent $\mathfrak{S}\prod_k^K \times_k U_k$. Moreover, $\mathfrak{S}$ is the identity tensor, $\mathfrak{S} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_K}$. Then function η can be described as $\eta = \|\tilde{\chi}\mu\mu^T \tilde{\chi}^T\|_F^2$. Here, let $y = \mu\mu^T$; thus $\eta = \|\tilde{\chi}y\tilde{\chi}^T\|_F^2$. We can obtain $\partial\eta/\partial\tilde{\chi} = \partial\|\tilde{\chi}y\tilde{\chi}^T\|_F^2/\partial\tilde{\chi} = [y\tilde{\chi}^T + y^T\tilde{\chi}^T]^T$ by using the derivation formula of Frobenius function. Therefore $\partial\eta/\partial\tilde{\chi} = 2(\tilde{\chi}\prod_k^K \times_k U_k)\prod_k^K \times_k U_k^T$.

Proof A.0.2. Using (10), we can get function $\gamma = \|\chi\prod_k^K \times_k C_k - \tilde{\chi}\prod_k^K \times_k U_k\|_F^2$. We use λ to represent $\chi\prod_k^K \times_k C_k$ and use μ to represent $\prod_k^K \times_k U_k$. Then function γ can be written as $\gamma = \|\lambda - \tilde{\chi}\mu\|_F^2$. We declare a function $tr, tr(\omega^T\omega) = \sum_{i_1}^{I_1} \sum_{i_2}^{I_2} \dots \sum_{i_N}^{I_N} \omega(i_1, i_2, \dots, i_N)^2$ for any $\omega \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$. For any matrix A , $\|A\|_F = \sqrt{tr(A^T A)}$. Then we can obtain $\gamma = tr(\lambda^T \lambda) - 2tr(\mu^T \tilde{\chi}^T \lambda) + tr(\mu^T \tilde{\chi}^T \tilde{\chi} \mu)$. Since $tr(\mu^T \tilde{\chi}^T \lambda) = tr(\lambda \mu^T \tilde{\chi}^T)$, $tr(\mu^T \tilde{\chi}^T \tilde{\chi} \mu) = tr(\tilde{\chi} \mu \mu^T \tilde{\chi}^T)$, $\partial\gamma/\partial\tilde{\chi} = \partial tr(\lambda^T \lambda)/\partial\tilde{\chi} - 2(\partial tr(\lambda \mu^T \tilde{\chi}^T)/\partial\tilde{\chi}) + \partial tr(\tilde{\chi} \mu \mu^T \tilde{\chi}^T)/\partial\tilde{\chi}$. Using Lemma 11, $\partial\gamma/\partial\tilde{\chi} = -2\lambda\mu^T + 2\tilde{\chi}\mu\mu^T$. When $\partial\gamma/\partial\tilde{\chi} = 0$, $\tilde{\chi} = \lambda\mu^T = (\chi\prod_k^K \times_k C_k)\prod_k^K \times_k U_k^T$. Finally, we can obtain (11) which is the optimal solution of (10) in Problem 10 while factor matrices $\{C_k\}$ and $\{U_k\}$ satisfy Conditions 1 and 2.

B.

Proof B.0.3 gives the proof of Lemma 13. Combining Lemma 13 and the ALS algorithm, Proof B.0.4 gives the solution of Problem 9.

Proof B.0.3. The target function is δ , where $\delta = tr((\chi\prod_k^K \times_k C_k)^T (\chi\prod_k^K \times_k C_k))$. By expanding the above formula, we can obtain that $\delta = tr((\chi\prod_{k \neq k_0}^K \times_k C_k \times_{k_0} C_{k_0})^T (\chi\prod_{k \neq k_0}^K \times_k C_k \times_{k_0} C_{k_0}))$. We use ℓ to represent $\chi\prod_{k \neq k_0}^K \times_k C_k$. Then function δ is described as $\delta = tr((\ell \times_{k_0} C_{k_0})^T (\ell \times_{k_0} C_{k_0}))$. The above-mentioned equation is abbreviated as $\delta = tr((\ell C_{k_0})^T (\ell C_{k_0}))$. Therefore, $\partial\delta/\partial C_{k_0} = \partial tr((\ell C_{k_0})^T (\ell C_{k_0}))/\partial C_{k_0} = \partial tr(C_{k_0}^T \ell^T \ell C_{k_0})/\partial C_{k_0}$. Let $y_0 = \ell^T \ell$; then we can obtain $\partial\delta/\partial C_{k_0} = \partial tr(C_{k_0}^T y_0 C_{k_0})/\partial C_{k_0}$. By using the derivation formula of Frobenius function, we conclude that $\partial\delta/\partial C_{k_0} = [y_0 C_{k_0} + y_0^T C_{k_0}]$; that is, $\partial\delta/\partial C_{k_0} = 2y_0 C_{k_0}$. Finally $\partial\delta/\partial C_{k_0} = \varepsilon(\chi\prod_{k \neq k_0}^K \times_k C_k)^T (\chi\prod_{k \neq k_0}^K \times_k C_k)$.

Proof B.0.4. We declare a function $\varsigma = \phi^2 = \|\chi\prod_k^K \times_k C_k - \nu\prod_k^K \times_k U_k\|_F^2$ using (8). If $\{C_k\}$ and $\{U_k\}$ can obtain the minimum of ς , then they can also generate the minimum of ϕ . We calculate the partial derivative of function ς with respect to C_{k_0} while fixing $\{C_k\}$ where $k \neq k_0$ and $\{U_k\}$. We use ℓ_0 to represent $\chi\prod_{k \neq k_0}^K \times_k C_k$ and ν_0 to represent $\nu\prod_k^K \times_k U_k$. Function ς can be written as $\varsigma = \|\ell_0 \times_{k_0} C_{k_0} - \nu_0\|_F^2$. We can abbreviate it as follows. $\varsigma = \|\ell_0 C_{k_0} - \nu_0\|_F^2$,

$$\begin{aligned} \frac{\partial \varsigma}{\partial C_{k_0}} &= \frac{\partial \|\ell_0 C_{k_0} - \nu_0\|_F^2}{\partial C_{k_0}} \\ &= \frac{\partial (tr(\nu_0^T \nu_0) - 2tr(C_{k_0}^T \ell_0^T \nu_0) + tr(C_{k_0}^T \ell_0^T \ell_0 C_{k_0}))}{\partial C_{k_0}} \end{aligned} \quad (B.1)$$

Using Lemma 13, we can obtain $\partial\varsigma/\partial C_{k_0} = -2\ell_0^T \nu_0 + 2\ell_0^T \ell_0 C_{k_0}$. We set $\partial\varsigma/\partial C_{k_0}$ to zero. We can get that $\ell_0 C_{k_0} = \nu_0$. Since C_{k_0} meets (9), we declare a temporary variable G_{k_0} . Let

$$G_{k_0} = (\ell_0^T)_{k_0} (\nu_0)_{k_0} \quad (B.2)$$

By using the abovementioned formula, we obtain the transformation $G_{k_0} = C_{k_0} G_{k_0}^T C_{k_0}$. We decompose matrix G_{k_0} using SVD

$$G_{k_0} = P_{k_0} S_{k_0} Q_{k_0}^T \quad (B.3)$$

where P_{k_0} and Q_{k_0} are the orthogonal matrices, and S_{k_0} is diagonal matrix with nonnegative elements. And eventually we can determine

$$C_{k_0} = P_{k_0} Q_{k_0}^T \quad (B.4)$$

and $\partial\varsigma/\partial U_{k_0}^T$ is calculated similarly to $\partial\varsigma/\partial C_{k_0}$.

C.

Proof C.0.5 provides the proof of Lemma 14.

Proof C.0.5. On the basis of the total and indirect derivative rules, we can obtain that $\partial lossF/\partial \nu = \sum_n^N ((\partial lossF/\partial \tilde{\chi}^{(n)}) (\partial \tilde{\chi}^{(n)}/\partial \nu))$. Furthermore, we can obtain $\partial \tilde{\chi}^{(n)}/\partial \nu = \partial [(\chi^{(n)} \prod_k^K \times_k C_k^{(n)}) \prod_{k_1}^K \times_{k_1} U_{k_1}^{(n)T}]/\partial \nu$ using (11). In accordance with (B.2), (B.3), and (B.4), we determine that $\{C_k^{(n)}\}$ and $\{U_{k_1}^{(n)T}\}$ are the functions of ν . Then we can obtain (13) and (14) by calculating the differentials $\partial \tilde{\chi}^{(n)}/\partial C_k^{(n)}$, $\partial C_k^{(n)}/\partial \nu$, $\partial \tilde{\chi}^{(n)}/\partial U_{k_1}^{(n)T}$, and $\partial U_{k_1}^{(n)T}/\partial \nu$. Finally we derive (12) which shows the method for calculating $\partial lossF/\partial \nu$.

Data Availability

The users' legal consulting statement data after processing used to support the findings of this study are currently under embargo while the research findings are commercialized. Requests for data, [6/12 months] after publication of this article, will be considered by the corresponding author.

Conflicts of Interest

The researchers claim no conflicts of interests.

Acknowledgments

In the research and writing work of this article, we would like to thank teachers and students of our lab team (Network and

Information Security Research Center, Harbin Institute of Technology) for their selfless help. This work was supported by the National Key Research and Development Program of China (2018YFC0830602, 2016QY03D0501).

References

- [1] K. Kirkpatrick, "Legal advice on the smartphone," *Communications of the ACM*, vol. 59, no. 7, pp. 19–21, 2016.
- [2] B. M. Gaff, S. G. Huggard, and G. W. Carey, "Do i need a lawyer? if you have to ask, you probably do," *The Computer Journal*, vol. 45, no. 5, pp. 10–12, 2012.
- [3] R. Yaroshinsky, R. El-Yaniv, and S. S. Seiden, "How to better use expert advice," *Machine Learning*, vol. 55, no. 3, pp. 271–309, 2004.
- [4] A. L. Tyree, G. Greenleaf, and A. Mobray, "Generating legal arguments," *Knowledge-Based Systems*, vol. 2, no. 1, pp. 46–51, 1989.
- [5] S. Gupta, S. Kamali, and A. Lpez-Ortiz, "On the advice complexity of the k -server problem under sparse metrics," *Theory of Computing Systems*, vol. 8179, no. 3, pp. 1–24, 2013.
- [6] A. K. Uysal, "An improved global feature selection scheme for text classification," *Expert Systems with Applications*, vol. 43, no. C, pp. 82–92, 2016.
- [7] O.-M. Sulea, M. Zampieri, S. Malmasi, M. Vela, L. P. Dinu, and J. Van Genabith, "Exploring the use of text classification in the legal domain," in *Proceedings of the 2nd Workshop on Automated Semantic Analysis of Information in Legal Texts, ASAIL 2017*, 2017.
- [8] I. S. Bajwa, F. Karim, M. A. Naeem, R. U. Amin et al., "A semi supervised approach for catchphrase classification in legal text documents," *Journal of Computers*, vol. 12, no. 5, 2017, p. 451+. Academic OneFile, Accessed 6 Mar. 2019.
- [9] F. Galgani and A. Hoffmann, "Lexa: Towards automatic legal citation classification," in *Proceedings of the Ai 2010: Advances in Artificial Intelligence - Australasian Joint Conference*, pp. 445–454, Adelaide, Australia, December 2010.
- [10] X. Xiong, "Application of quadratic dimension reduction method based on lsa in classification of the chinese legal text," *Electronic Measurement Technology*, 2007.
- [11] E. De Maat, K. Krabben, and R. Winkels, "Machine learning versus knowledge based classification of legal texts," *Frontiers in Artificial Intelligence and Applications*, vol. 223, pp. 87–96, 2010.
- [12] R. Bartolini, A. Lenci, S. Montemagni, V. Pirrelli, and C. Soria, "Automatic classification and analysis of provisions in Italian legal texts: a case study," in *On the Move to Meaningful Internet Systems 2004: OTM 2004 Workshops: OTM Confederated International Workshops and Posters, GADA, JTRES, MIOS, WORM, WOSE, PhDS, and INTEROP 2004, Agia Napa, Cyprus*, vol. 3292 of *Lecture Notes in Computer Science*, pp. 593–604, Springer, 2004.
- [13] P. Wang, B. Xu, J. Xu, G. Tian, C.-L. Liu, and H. Hao, "Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification," *Neurocomputing*, vol. 174, pp. 806–814, 2016.
- [14] K. Nigam, A. K. Mccallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Machine Learning*, vol. 39, no. 2, pp. 103–134, 2000.
- [15] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, Doha, Qatar, October 2014.
- [16] J. Donahue, L. A. Hendricks, S. Guadarrama et al., "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 2625–2634, USA, June 2015.
- [17] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014.
- [18] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.
- [19] M. Nakatsuji, H. Toda, H. Sawada, J. G. Zheng, and J. A. Hendler, "Semantic sensitive tensor factorization," *Artificial Intelligence*, vol. 230, no. C, pp. 224–245, 2016.
- [20] A. H. Phan and A. Cichocki, "Tensor decompositions for feature extraction and classification of high dimensional datasets," *Nonlinear Theory and Its Applications, IEICE*, vol. 1, no. 1, pp. 37–68, 2010.
- [21] A. Bernardi, J. Brachat, P. Comon, and B. Mourrain, "General tensor decomposition, moment matrices and applications," *Journal of Symbolic Computation*, vol. 52, no. 9, pp. 51–71, 2013.
- [22] Y. H. Taguchi, "Tensor decomposition-based unsupervised feature extraction applied to matrix products for multi-view data processing," *Plos One*, vol. 12, no. 8, Article ID e0183933, 2017.
- [23] I. V. Oseledets, "Tensor-train decomposition," *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, 2011.
- [24] Y.-H. Taguchi, "Identification of candidate drugs using tensor-decomposition-based unsupervised feature extraction in integrated analysis of gene expression between diseases and drug-matrix datasets," *Scientific Reports*, vol. 7, no. 1, 2017.
- [25] M. V. Rakhuba and I. V. Oseledets, "Grid-based electronic structure calculations: the tensor decomposition approach," *Journal of Computational Physics*, vol. 312, pp. 19–30, 2016.
- [26] Y. Wu, H. Tan, Y. Li, F. Li, and H. He, "Robust tensor decomposition based on Cauchy distribution and its applications," *Neurocomputing*, vol. 223, pp. 107–117, 2017.
- [27] T. Chen, R. Xu, Y. He, and X. Wang, "Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN," *Expert Systems with Applications*, vol. 72, pp. 221–230, 2017.
- [28] L. O. Chua and T. Roska, "CNN paradigm," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 3, pp. 147–156, 1993.
- [29] K. Cho, B. V. Merriënboer, C. Gulcehre et al., "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *Computer Science*, 2014.
- [30] T. Bansal, D. Belanger, and A. McCallum, "Ask the gru: multi-task learning for deep text recommendations," in *Proceedings of the 10th ACM Conference on Recommender Systems, RecSys 2016*, pp. 107–114, USA, September 2016.
- [31] S. B. Kakvaeva, "Substantive word combinations with dependent adjectives, pronouns, numerals, participles and adverbs in russian and lak languages," *Sociosfera*, no. 4, 2012.

- [32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *Computer Science*, 2013.
- [33] Y. Goldberg and O. Levy, "Word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method," <https://arxiv.org/abs/1402.3722>.