A Novel Convolutional Neural Network for Statutes Recommendation

Chuanyi Li[1,2], Jingjing Ye[2], Jidong Ge[1,2(✉)], Li Kong[2], Haiyang Hu[1,3], and Bin Luo[1,2]

[1] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
lcynju@l26.com, gjdnju@l63.com
[2] Software Institute, Nanjing University, Nanjing, China
[3] School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China

Abstract. In recent years, statutes recommendation has been a popular research subject of artificial intelligence in legal domain. However, the existing statutes recommendation systems are more oriented to professionals, such as judges and lawyers, and are not suitable for general public who have no legal knowledge and cannot independently extract key points. We use deep learning to solve the ambiguity and variability of general public's linguistic expressions about cases. We propose a novel Convolutional Neural Network (CNN) architecture to obtain the relations between statutes and cases. Unlike previous works, in order to utilize the semantics of statutes, we also put statute content as model input besides case description. Moreover, different from the *Top-k* method, the numbers of statutes recommended by our model varies among cases. In addition, all the features of the case statements and statute contents are extracted automatically without any human intervention. So, the approach for training the model can be easily applied in different types of cases and laws. Experiments results on the juridical document corpus of the proposed CNN model surpass those of previous neural network competitors.

Keywords: Statutes recommendation · Convolutional Neural Network
Natural Language Processing

1 Introduction

As artificial intelligence is applied in more and more applications, courthouses are starting to focus on intelligent judges. Robot judges, robot legal consultants and other products are emerging. Statutes recommendation is an important part of judicative intelligence. Because statutes are the support of case verdicts. If we can predict the statutes accurately, we can get the trends of verdicts to some extent. What's more, recommending proper statutes for cases is quite useful for all roles involved in legal cases, such as judges, lawyers and interested parties. It can help the judges to process the cases more effectively and efficiently. It can also impel lawyers to find more references so as to defend better. For people without professional legal knowledge,

X. Geng and B.-H. Kang (Eds.): PRICAI 2018, LNAI 11012, pp. 851–863, 2018.
https://doi.org/10.1007/978-3-319-97304-3_65

statutes recommendation is much more helpful. It is hard for them to find proper statutes without the assistance of professionals. Seeking advice from law firms costs much time and money. A system which can recommend proper statutes according to given case descriptions will benefit them a lot.

There have been a few studies on statutes prediction or recommendation (Kim et al. 2016; Chen and Chi 2010; Chou and Hsing 2010; Conrad and Schilder 2007; Moens 2001). However, most of them focused on retrieving or classifying statutes based on keywords, which are difficult to use for people without professional legal knowledge. The recent improvement made by Liu et al. (2015) considers to retrieve relevant statutes from a query sentence using daily customary terms. Liu et al. (2015) implemented a system to classify user query by Support Vector Machine (SVM). The accuracy of this model largely depends on the quality of the input query sentences. However, key points of a case cannot always be summarized within a proper chief query sentence especially by non-professional users. They may describe a case by some facts which are irrelevant to the final judgment. Besides, the model always returns a fixed number (e.g., 5 or 10) of statutes for any case, although the number of statutes cited by cases varies a lot, which is from one to even over twenty. This is also a potential limitation of most of existing statutes recommendation systems. Furthermore, the adopted multi classification strategy is sensitive to the category distribution of training samples, and categories imbalance may lead to the deviation of model. The classifier model is more inclined to recommend popular statues and ignore the infrequent statutes directly. But popular statutes are usually universal to many cases. For example, the 64-th article in *Civil Procedure Law of the People's Republic of China* reads: "当事人对自己提出的主张，有责任提供证据……" (It is the duty of a party to an action to provide evidence in support of his allegations…[1]). These popular statutes are usually not the crucial statutes which affect the judgement of cases. The users pay more attention to the statutes which are closely related to the cases. Lastly, most of existing systems do not use the statutes' semantic information to extract the case features more accurately. They just regard statutes as labels.

In this paper, we propose a statutes recommendation framework based on a novel CNN model to overcome existing problems. There are three key features of the pro-posed framework:

1. One of the inputs is plaintiff claiming segment using daily customary terms. It is the case description submitted to the court by the plaintiff, which does not contain many professional legal terms. This ensures the usability of system for general public. At the same time, it is the only information that judges and lawyers can get before hold hearings.
2. Statute contents are also treated as the inputs of the model. This is beneficial to highlight the crucial features of cases and reduce the weight of irrelevant information. It can also assist us in finding statutes more relevant to cases.
3. The statutes predicting problem is formalized as a binary classification task: determine whether the statute is suitable for the case according to the given text

[1] The English version of Civil Procedure Law of the People's Republic of China, http://www.npc.gov.cn/englishnpc/Law/2007-12/12/content_1383880.htm.

couple of statute and case. Compared to the pervious classifying model which only extracts the features of cases, the goal of our CNN model is to capture the semantic relations between case description and the reference statutes.

The flow chart of the model is shown in Fig. 1. If the number of candidate statutes is *k*, we should run this classifier model for k times. So, the numbers of recommended statutes will not affect each other. In real applications, we can limit the number of candidate statutes to a reasonable number by restraining the cause of cases or requesting user select law scope.

The rest of this paper is organized as follows. Section 2 presents the research background, including recommender systems (RS), CNN and related techniques, as well as discussion of the motivation of proposing a novel CNN model for predicting statutes. Section 3 introduces the proposed CNN architecture in detail. The experiments and results are presented in Sect. 4. Section 5 concludes the paper.

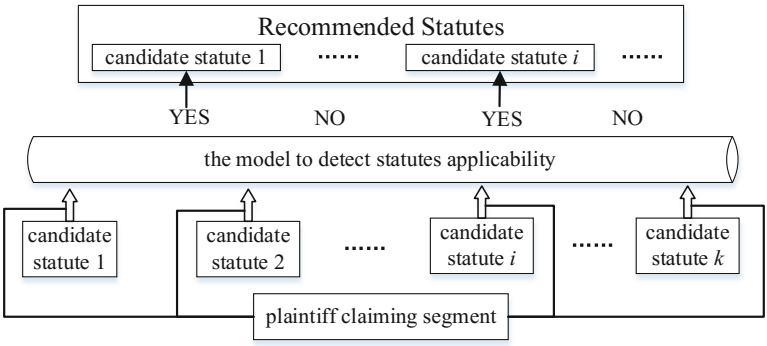

Fig. 1. The flow chart of the statutes recommendation model

2 Background

2.1 Recommender System

Recommender Systems are software tools to assist users in finding useful information quickly. The term first appeared in the 1990s (William et al. 1995; Shardanand et al. 1995; Resnick et al. 1994). Unlike the search engine, the recommender systems don't require the user to provide clear requirements. They analyze the users' historical behaviors and model the users' interests so as to recommend information that meets users' demands. Nowadays, recommender systems are applied in a diversity of fields including music, books, finance, law etc. The recommender system algorithms can be mainly classified into three categories: collaborative filtering, content-based filtering and hybrid methods.

In the scene of statutes recommendation, each case needs to be recommended only once. They don't have behavior histories. Every recommendation is faced to a "new user". In the early stage of the study, we have tried the filtering based on the

neighborhood. We found out the similar cases through the case features and used the similar cases' statutes as the recommended statutes. But, unfortunately, the result of the experiment was bad. We analyzed the data set and found that the case statements of the same case cause are pretty similar. For example, the statement of a divorce case is "原告李娟诉称，2001年6月，原、被告经人介绍相识。后于2002年3月10日登记结婚。婚后未有子女。由于双方婚前缺乏了解，婚后感情不和，经常为生活琐事生气，夫妻感情已经完全破裂。现请求依法判令解除原、被告的婚姻关系。" (The plaintiff Li Juan said that, the plaintiff and the defendant introduced to each other in June 2001, and then they got married in March 10, 2002. There is no child after marriage. Due to the lack of understanding before marriage, they are on bad terms and always argue for trivial matters after marriage. The affection between husband and wife is shattered entirely. Now the plaintiff askes the court to release the marriage relationship between the plaintiff and the defendant.) The statement of another case is almost the same except for having a son and a daughter after marriage. The intersection of statutes cited by the two cases is empty. Obviously, whether having a child or not is a key point. But it is very difficult for machine to learn key elements from a large number of similar parts. We can list the key points manually, but the cost is too high. Just for divorce cases, there are also many key points such as family violence, property disputes, paramour, bigamy and so on. For thousands of case causes, the workload is too heavy.

So, we decide to adopt content-based filtering. We add statute features as inputs in order to highlight the key parts in the case statements. We predict by analyzing the relationship between case statements and statute contents.

2.2 Convolutional Neural Network

CNN model has its success on fields like computer vision (Neverova et al. 2014), speech recognition (Deng et al. 2013) and natural language processing (Collobert and Weston 2008). A filter with width m can learn to recognize specific n-grams of texts where n is less than or equal to the filter width m. What's more, the position of n-grams hardly influences the meaning of the sentence. So, pooling operation which owns a property of local translation invariance can help to capture features more effectively. Previous studies have shown that CNN model has a good performance on text classification, sentiment analysis, and text similarity (Kim 2014; Kalchbrenner et al. 2014; He et al. 2015). Accordingly, we attempt to use a CNN model to recommend statutes for cases.

2.3 Attention Mechanism

Attention mechanism was first widely used in computer vision (Mnih et al. 2014), and was extended to machine translation in Natural Language Processing (NLP) field by Bahdanau et al. (2015). It is used to find out the words in the source language related to the generated word in the target language. As a result, translating as well as aligning will be done at the same time. Attention mechanism can observably improve the accuracy of the translation. It is similar to statutes recommendation since we want to find out the key words of case statements related to the statute, just like the alignment in

machine translation. Inspired by attention mechanism, we adopt a correlation matrix to measure semantic similarity between statutes and cases in our model.

2.4 Word Vectors

Word vectors generated by neural network language model were proposed by Bengio et al. in 2003. Word vectors are used to reconstruct the representation of words into form of vectors. Compared to TF-IDF, it has a nice property that semantical close words are likewise close in Euclidean or cosine distance. This model is suitable for various languages, including Chinese. In our experiment, we used word vectors to represent words in lower dimensional vector space. We constructed multiple correlation matrices by using different distance measurements.

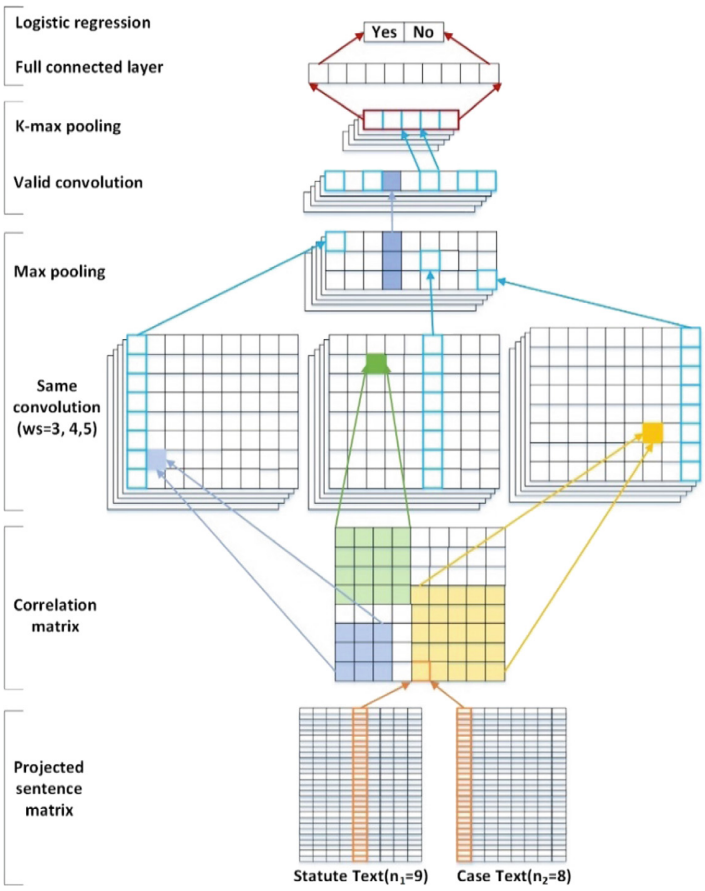

Fig. 2. The CNN architecture to judge whether the statute is suitable for certain case

3 The Proposed CNN Structure

In this section, we describe our model which is shown in Fig. 2. The model is used to judge whether the statute is suitable for certain case by means of measuring semantic relations between the statute and the basic situation of the case.

3.1 Correlation Matrix

We combine case statements and statute contents at the first layer of the network in order to capture their relations. Our inputs are pairs of texts, which have distinct lengths. Let S^1 represents the statute text and S^2 represents the case text. An embedding layer which can be fine-tuned during training will project each word of the text to a q-dimensional word vector. Suppose that S_i^1 responds to the vector of the i-th word in S^1, then the dot correlation matrix $M^{dot} \in R^{n_1 \times n_2}$ can be represented as

$$M_{ij}^{dot} = S_i^1 \cdot S_j^2, \tag{1}$$

where n_1 is the length of S^1 and n_2 is the length of S^2. There are other alternative correlation matrices, for example, Euclidean correlation matrix M^{Euc}, shown in Eq. (2) and Manchester correlation matrix M^{Man}, shown in Eq. (3).

$$M_{ij}^{Euc} = \sum_{d=1}^{q} \left(S_{i,d}^1 - S_{j,d}^2 \right)^2 \tag{2}$$

$$M_{ij}^{Man} = \sum_{d=1}^{q} \left| S_{i,d}^1 - S_{j,d}^2 \right| \tag{3}$$

Here $S_{i,d}^1$ means the d-th dimension of S_i^1. We can also use multiply matrices by concentrating them into multi-channel, which can improve performance slightly, but do harm to efficiency.

3.2 Convolution

The convolution operation at the first convolution layer of the network is convolving a matrix of weights $W_c \in R^{ws \times ws \times c}$ with correlation matrices mentioned above, where ws is the window size and c is the number of correlation matrices we applied. We use Same Convolution (Fukushima and Neocognitron 1982) with zero padding to control the kernel width and the size of the output independently and gain the output of the same size with different window sizes. After that, each value in the output matrix should be added to a bias and perform an operation of a nonlinear function, such as ReLU.

A feature map can be seen as extracting a certain feature of the input. We use multiple feature maps so that we can gather different kinds of text features, which is helpful to improve the system performance.

Denoting the number of feature maps as f_s, the output of Same Convolution has size $n_1 \times n_2 \times f_s$. We pick up the maximum value over dimension n_2 through a

max-overtime pooling operation (Collobert et al. 2011). The resulting matrix has dimensions $n_1 \times f_s$.

We apply multiply window sizes with same filter map size and concentrate the outputs together. We denote the generating matrix as O^s whose dimensions are $n_1 \times g \times f_s$, where g is the number of distinct window sizes. Then, we convolute weights $U \in R^{1\times g\times f_s}$ with O^s at the second convolution layer. Similarly, there are more than one feature maps and each feature map has a bias term and a nonlinear function. However, we use Valid Convolution (Fukushima and Neocognitron 1982) without zero padding rather than Same Convolution this time.

3.3 *K*-max Pooling

Please notice that the number of input neurons at the full connected layer is fixed, but the lengths of S^1 and S^2 vary considerably. In order to solve the problem of variable text lengths, we use Kalchbrenner et al. (2014)'s *k*-max pooling to project sentences of distinct lengths to the same size. Unlike max pooling, it selects *k* top values from each dimension rather than one. As a result, it avoids the loss of features. For example, max pooling can't distinguish whether a valuable feature occurs one or multiple times in a single row.

3.4 Full Connection

The output matrix of *k*-max pooling layer will be flattened to a vector v as the input of full connected layer. Then, the vector v will be multiplied with weights W_f and be added to a scalar b_f. The result will be operated by sigmoid function to get the probability whether the statute is suitable to the case. It can be formulated as

$$y = \sigma\left(W_f v + b_f\right). \tag{4}$$

The whole operation can also be seen as a logistic regression.

We train the network by minimizing the cross-entropy loss between the predicted and expected distributions. We also employ dropout and L2 regularization on the weight and bias vectors at the penultimate layer to prevent overfitting.

4 Experiments and Results

4.1 Datasets

We selected 13000 juridical documents of divorce cases randomly from China Judgment Documents Repository as the whole cases data set. The *juridical document* is a summary of the case, written by the judge after the completion of the trial. It includes the basic situation of the case, the parties, the evidences, the judgment result, the trial and analysis process and so on. It is used to document a case. Except for confidential cases, the juridical document is open to all, so it has high readability. We picked up the

contents of plaintiff claiming segment as case statements. The data set was randomly split into 4936 training, 1737 development and 6327 testing.

Unlike English, Chinese has the characteristics of continuous writing without blank characters. If the computer can't obtain the exact boundary of words, it is difficult to gain the semantic information contained in the text (Lai et al. 2013). So, word segment is a common data preprocessing operation in Chinese language processing. Nowadays, there are many excellent word segment tools, such as ANSJ, JIEBA, ICTCLAS, SCWS, LTP, NLPIR and so on. We used ANSJ to segment texts. ANSJ owns four segment patterns: base mode, precise mode, index mode and NLP mode. In this paper, we chose the NLP mode to segment word. Because it is the most accuracy mode and it supports digit recognition, name recognition, organization recognition and new word detection.

After word segment, we removed the following characters to filter out irrelevant interferential words.

- Words of time, place, people name, organization name were ignored.
- Prepositions and conjunctions were removed since their major function is to connect the grammatical structure, not to express semantic.
- We also removed non-Chinese characters and single character words since a Chinese word usually consists of at least two characters.
- The words appeared over 10000 times or less than 5 times in the whole corpus were discarded. We can't identify one case from all the cases through frequent words since most of case statements have these words. We also can't gather a group of cases cited the same statute by rare words since most of case statements don't own these words. So, both the frequent words and the rare words can't be-come the case features.
- Furthermore, we make a constraint that each word should only occur once in every 5-gram of the text. If there are more than one, only the first will be kept. Because we found that the beginning of a sentence in plaintiff claiming is often the repeat of last sentence. For example, "2003年7月，两人经人介绍相识。两人相识后，于2005年5月20日登记结婚。" (In July 2003, the plaintiff and the defendant introduced to each other. After acquainted with each other, they married in May 20, 2005.). The reason of choosing number 5 is that the max window size of our CNN model is 5.

The length of case statement inputs varies from 10 to 317. The cases distribution based on the length of preprocessed case statements is shown in Table 1.

Table 1. The cases distribution based on the length of preprocessed case statements.

Dataset	10–19	20–29	30–39	40–49	50–59	60–69	70–79	80–89	≥90
Training	731	1346	996	704	418	245	127	82	287
Test	975	1696	1421	940	535	282	230	116	132

We only chose the whole 50 statutes in *Marriage Law of the People's Republic of China* as the candidate statutes since statutes in the same law are more similar and

confusing. For example, the following three statutes in *Marriage Law of the People's Republic of China*[2] are quite analogous.

- 第三条 禁止包办、买卖婚姻和其他干涉婚姻自由的行为。禁止借婚姻索取财物。禁止重婚。禁止有配偶者与他人同居。禁止家庭暴力。禁止家庭成员间的虐待和遗弃。(Article 3 Marriage upon arbitrary decision by any third party, mercenary marriage and any other acts of interference in the freedom of marriage shall be prohibited. The exaction of money or gifts in connection with marriage shall be prohibited. Bigamy shall be prohibited. Anyone who has a spouse shall be prohibited to cohabit with another person of the opposite sex. Family violence shall be prohibited. Maltreatment and desertion of one family member by another shall be prohibited.)
- 第十条 有下列情形之一的，婚姻无效：（一）重婚的；（二）有禁止结婚的亲属关系的；（三）婚前患有医学上认为不应当结婚的疾病，婚后尚未治愈的；（四）未到法定婚龄的。(Article 10 The marriage shall be invalid if: (1) either of the married parties commits bigamy; (2) there is the prohibited degree of kinship between the married parties; (3) before marriage either of the parties is suffering from a disease which is regarded by medical science as rending a person unfit for marriage and which has not yet been cured after marriage; or (4) one of the married parties has not reached the statutory age for marriage.)
- 第四十六条 有下列情形之一，导致离婚的，无过错方有权请求损害赔偿：（一）重婚的；（二）有配偶者与他人同居的；（三）实施家庭暴力的；（四）虐待、遗弃家庭成员的。(Article 46 Where one of the following circumstances leads to divorce, the unerring party shall have the right to claim compensation: (1) bigamy is committed; (2) one party who has a spouse cohabits with another person of the opposite sex; (3) family violence is committed; or (4) a family member is maltreated or abandoned.)

If the model can choose proper statutes from these similar statutes, it should be easier to distinguish less similar statutes and get better results. If you want to get the statute contents of the whole laws, you may pay to the courts or the law agencies who own the law database. The statute contents were preprocessed as the same as case statements. The length of statute inputs varies from 3 to 55. The cases distribution based on the number of quoted statutes is shown in Table 2.

Table 2. The cases distribution based on the number of quoted statutes.

Dataset	1	2	3	4	5	6	7	8	9
Training	2839	739	738	400	157	42	14	5	2
Test	3596	1003	944	483	217	65	15	4	0

[2] The English version of Marriage Law of the People's Republic of China, http://www.npc.gov.cn/englishnpc/Law/2007-12/13/content_1384064.htm.

The distribution of the positive and negative classes is extremely unbalanced, since statutes unquoted by a case are far more than quoted. In order to avoid the negative class shift, we randomly selected four unquoted statutes as negative samples for each positive sample. The cited frequency of the statute determines the probability of the selection. The training sample format is (statute content, case statement, 1) for positive sample and (statute content, case statement, 0) for negative sample.

4.2 Pre-trained Word Vectors

Pre-training word vectors from a large corpus is a common method in natural language processing. It can improve performance in the absence of a large supervised training set (Collobert et al. 2011; Socher et al. 2011; Iyyer et al. 2014).

We trained the word vectors from all the juridical documents of cases happened in 2015 and 2016, totally 2000000 pieces. We used the full text of the juridical documents, rather than plaintiff claiming segment. We still selected ANSJ as the word segmentation tool and adopted the NLP mode. We removed all the non-Chinese characters and single character words. The vectors have dimensionality of 200 and were trained using the continuous bag-of-words architecture (Le and Mikolov 2014). Words not included in the set of pre-trained words are initialized to be mean of all word vectors.

4.3 Training

Training is done through stochastic gradient descent over shuffled mini-batches of size 64 with the Adagrad update rule (Duchi et al. 2011).

We use rectified linear units as the activation function. Filter window sizes of Same Convolution are 3, 4, 5 and each has 128 feature maps while Valid Convolution has 256 feature maps. Dropout rate is 0.5, L2 regularization weight is 5×10^{-4}, and learning rate is 10^{-3}.

We chose two approaches as contrasts. The one is the CNN model proposed by Kim (2014) which has been proved to be one of the best models on text classification. The case statements are inputs and the statute labels are outputs. We consider each candidate statute as a label, so the number of output neurons is 50. We choose the top-2 statutes as a recommendation. Since the network cannot handle the inputs with random lengths, we regard the largest text length in the training set as the fixed text length. If the length of the input is smaller than that, pad with <unk>. If it is larger than that, it will be truncated.

The other contract model is the CNN model proposed by He et al. (2015) which has superior performance on text similarity. Both case statements and statute contents are inputs while outputs are binary classes. Notice that it cannot handle the inputs with random lengths, too. Text should be padded or truncated to a fixed length as above.

4.4 Results

In this paper, we use precision, recall and F1-score to measure the model performance. Precision is a measure of the correctness of the recommendation. It is defined as:

$$Precision = \frac{1}{N}\sum\frac{R}{T}, \tag{5}$$

where N denotes the total number of cases, R denotes the number of statutes which are predicted right, T denotes the total number of statutes which are recommended. Recall is a measure of the coverage of the recommendation. The formula is as follows:

$$Recall = \frac{1}{N}\sum\frac{R}{S}, \tag{6}$$

where S is the number of statutes which are actually cited. Precision and recall are mutual condition. For one extreme example, if we recommend all candidate statutes, the recall will be 100% and the precision will be quite low. So, we can use F1-score as a compromise. F1-score is the harmonic mean of precision and recall. It is defined as:

$$F1\text{-}score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{7}$$

Table 3 shows the experiment results on divorce cases. Our model outperforms the other systems. Via the results of Kim's CNN model and ours, it embodies the key insight that the semantic information of statute contents is beneficial to identifying statutes applicability, which is agreed with our expectation. The possible reason for surpassing the approach of He et al. (2015) is that we put the case statements and statute contents together at the first layer of CNN model while He et al. (2015) associates them until the last full connected layer.

Table 3. Experiment results on divorce cases.

Model	Precision	Recall	F1 score
Kim 2014	0.6321	0.8090	0.7069
He et al. 2015	0.8749	0.7023	0.7791
Ours	0.9359	0.7173	0.8121

5 Conclusion

In this paper, we presented a novel CNN model for statutes recommendation. We use word2vec to express the semantic meaning of the text. We combine statute contents and case statements at the first layer of the network. Then we obtain the relations between these two texts through the convolution of different kernel sizes and *k*-max pooling. In the end, whether the statute is used in the case or not is computed by the full connection layer. Experiments show that the model has good performance.

In future work, we will dedicate to make the proposed model achieve the ability to judge the applicability of new-released statutes, since the exiting statutes recommendation frameworks based on classification techniques cannot meet the demand. Besides, we will try to mine associative statute rules to optimize the outputs of our CNN model.

Acknowledgment. This work was supported by the National Key R&D Program of China (2016YFC0800803), the National Natural Science Foundation of China (No. 61572162, 61572251, 61702144), the Natural Science Foundation of Jiangsu Province (No. BK20131277), the Zhejiang Provincial Key Science and Technology Project Foundation (NO. 2018C01012), the Zhejiang Provincial National Science Foundation of China (No. LQ17F020003), and the Fundamental Research Funds for the Central Universities.

References

Kim, W., Lee, Y., Kim, D., Won, M., Jung, H.: Ontology-based model of law retrieval system for R&D projects. In: Proceedings of the 18th Annual International Conference on Electronic Commerce: e-Commerce in Smart Connected World, pp. 1–6 (2016)

Chen, C., Chi, J.Y.P.: Use text mining to generate the draft of indictment for prosecutor. In: Proceedings of the 2010 Pacific Asia Conference on Information Systems (PACIS), pp. 706–712 (2010)

Chou, S.C., Hsing, T.P.: Text mining technique for Chinese written judgment of criminal case. In: IEEE Intelligence and Security Informatics Conference, pp. 113–125 (2010)

Conrad, J.G., Schilder, F.: Opinion mining in legal blogs. In: Proceedings of the 11th International Conference on Artificial Intelligence and Law (ICAIL), pp. 231–236 (2007)

Moens, M.F.: Innovative techniques for legal text retrieval. In: Proceedings of the 5th International Conference on Artificial Intelligence and Law, pp. 29–57 (2001)

Liu, Y., Chen, Y., Ho, W.: Predicting associated statutes for legal problems. Inf. Process. Manag. **51**, 194–211 (2015)

Hill, W.C., Stead, L., Rosenstein, M., Furnas, G.W.: Recommending and evaluating Choices in a virtual community of use. In: The Proceedings of the 1995 International Conference of Human-Computer Interaction (CHI), pp. 194–201 (1995)

Shardanand, U., Maes, P.: Social information filtering: algorithms for automating "Word of Mouth". In: The Proceedings of the 1995 International Conference of Human-Computer Interaction (CHI), pp. 210–217 (1995)

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews (CSCW), pp. 175–186 (1994)

Neverova, N., Wolf, C., Taylor, G.W., Nebout, F.: Multi-scale deep learning for gesture detection and localization. In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8925, pp. 474–490. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-16178-5_33

Deng, L., Hinton, G., Kingsbury, B.: New types of deep neural network learning for speech recognition and related applications: an overview. In: Proceedings of 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8599–8603 (2013)

Collobert, R., Weston, J.: A unified architecture for natural language processing: deep neural networks with multitask learning. In: Proceedings of the 25th International Conference on Machine Learning, pp. 160–167 (2008)

Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1746–1751 (2014)

Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pp. 655–665 (2014)

He, H., Gimpel, K., Lin, J.: Multi-perspective sentence similarity modeling with convolutional neural networks. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1576–1586 (2015)

Mnih, V., Heess, N., Graves, A., Kavukcuoglu, K.: Recurrent models of visual attention. In Proceedings of the 2014 Annual Conference on Neural Information Processing Systems (NIPS), pp. 2204–2212 (2014)

Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: Proceedings of the 2015 International Conference on Learning Representations (ICLR), pp. 1–15 (2015)

Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C.: A neural probabilistic language model. J. Mach. Learn. Res. **2003**(3), 1137–1155 (2003)

Fukushima, K., Neocognitron, S.M.: A new algorithm for pattern recognition tolerant of deformations and shifts in position. Pattern Recogn. **15**(6), 455–469 (1982)

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**, 2493–2537 (2011)

Lai, S., Xu, L., Chen, Y., Liu, K., Zhao, J.: Chinese word segment based on character representation learning. J. Chin. Inf. Process. **2013**(5), 8–14 (2013)

Socher, R., Pennington, J., Huang, E., Ng, A., Manning, C.: Semi-supervised recursive autoencoders for predicting sentiment distributions. In: Proceedings of the 2011 International Conference on Empirical Methods in Natural Language (EMNLP), pp. 151–161 (2011)

Iyyer, M., Enns, P., Boyd-Graber, J., Resnik, P.: Political ideology detection using recursive neural networks. In: Proceedings of the 2014 Annual Meeting of the Association for Computational Linguistics (ACL), pp. 1113–1122 (2014)

Le, Q., Mikolov, T.: Distributed represenations of sentences and documents. In: Proceedings of the 2014 International Conference on Machine Learning (ICML), pp. 1188–1196 (2014)

Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. J. Mach. Learn. Res. **12**, 2121–2159 (2011)