# Predicting Appropriate Charges for Chinese Criminal Cases with Automatically Extracted Relevant Articles as Legal Basis


## Abstract

In this paper, we investigate the problem of determining the appropriate charges for a certain case under the circumstance of statutory law. This problem is important when users communicate with legal assistance systems with natural language, since we need to first determine the category of the case before giving any useful responses. We propose that judgement documents of statutory law countries like China provides natural training data for this task and we find that model multi-label document classification technics fit this problem well. Furthermore, we also find that using additional information of law articles can further improve the model performance in the system of statutory law. TODO: polish the abstract


## 1 Introduction

Determining appropriate charges (like *fraud*, *larceny*, and *homicide*) of a case is helpful for legal assistance systems when the user would like to query the system by describing the case, and it is even more important when the user has no idea of the legal basis of a case, where the only input he (or she) can give is the fact description of the case. For example, if the user wants to find similar cases, one can use the predicted charges of the query case to filter out irrelevant results. And if the user wants to know the possible penalties regarding a case, one also need to decide appropriate charges first.

In these situations, we need to predict appropirate charges based solely on the fact descriptions of a case. However, this charge prediction task is not easy: (1) The differences between two charges can sometimes be subtle. For example, in the context of criminal cases in China, distinguishing *intentional homicide* from *negligent homicide* would require detailed analysis of the behavior of the defendant. (2) Multiple crimes may be involved in a single case, which means we need to conduct charge prediction in the multi-label classification paradigm. (3) Although we can expect the model to implicitly learn the legal basis of the judgement through massive training data, the predicted charges are still not convincing enough if no law articles are involved in the prediction. This problem is prominent in countries using civil law system, e.g., China (except Hong Kong), where the judgement is made only based on statutory laws. For example, as shown in Figure 1, a judgement document in China always includes relevant law articles (usually in the court view part) to support the decision. Even in countries using common law system, e.g., the United States (except Louisiana), where the judgement is based mainly on decisions of previous cases, there are still some statutory laws that need to be followed when making decisions.

Previous works on charge prediction either use k-Nearest Neighbor (KNN) [Liu *et al.*, 2004; Liu and Hsieh, 2006] as classifier or need to manually design key factors of specific charges [Lin *et al.*, 2012], thus do not scale well with regard to data size and number of charges. Furthermore, none of them consider the situation where multiple charges are involved. On the other hand, [Liu and Liao, 2005; Liu and Hsieh, 2006] also try to find the specific law articles that has been violated, but converts the multi-label problem into multi-class classification by only considering a fixed set of article combinations, therefore does not scale to the situation when a larger set of articles are involved. [Liu *et al.*, 2015] aims to find relevant articles in a scalable way by doing predilinary classification first and rerank the results afterwards. Although the framework is promising, only shallow, i.e., word-level, semantic features are employed in their work. Furthermore, all of these works treat charge prediction and article prediction separately, ignoring the fact that they can actually benefit each other.

In our proposed method, however, we jointly model the charge prediction and relevant article extraction in a sinble framework, which enables them to influence each other in a positive way. Specifically, to understand the whole framework of the facts, inspired by previous works on document classification [Tang *et al.*, 2015; Yang *et al.*, 2016], we use a sentence-level and a document-level Gated Recurrent Unit (GRU) to model the associations among words and sentences. To capture the important details, we use attention mechanism to select the most informative words or sentences for sentence and document embedding respectively. To handle the multi-label nature of the problem, we convert the multi-label target to label distribution, and then use cross entropy as loss function. To find relevant law articles in the statutory laws, which contain a large number of articles, to support our charge prediction, we first use a simple BOW-based article classifier to quickly filter out most of the irrelevant articles.

Then we attentively aggregate the retained top $k$ articles for further charge classification.

We evaluate our method by predicting charges in Chinese criminal cases. Since the public judgement documents often contain the fact descriptions, relevant law articles and corresponding charges (see Figure 1), they actually provide natural large-scale high-quality trianing data for our task. We therefore use rules and regular expressions to extract these information and build a dataset accordingly. The experimental results on this dataset show that our method significantly outperforms the baselie BOW-based Suport Vecotr Machine (SVM) method, and the automatically extracted relevant law articles can clearly improve the model with only facts as input. We also find that our model can effectively attend to the true relevant articles, which to some extent provides us with the legal basis for our charge prediction. Furthermore, since there exists differences between the words used by laymen and legal practitioners, we also annotate 100 news to test the performance of our model on fact descriptions wrtitten by odinary people, which is more useful in practice. The experimental results show that, although trained on judgement document data, our model still has reasonable generalizatoin ability in this situation.

The contributions of this paper are threefold: (1) It proposes a novel neural network model that can jointly utilize the facts and the automatically extracted relevant law articles of a case to predict appropriate charges. (2) The proposed model can aslo provide legal basis for the charge prediction in the form of weighted relevant law articles. (2) By further evaluating the model on human labeled news data, it shows that the model trained on judgement documents have reasonable generalization ability on the text written by people who are not legal practitioners.

## 2 Related Work

In document classification, a simple but effective method is to combine bag-of-words (BOW) features with varies classifiers [Joachims, 1998]. Recently, neural network (NN) models like Convolutional Neural Network (CNN) [Kim, 2014] have been used for document embedding, and the resultant document vecotr can be further used for classification. [Tang *et al.*, 2015] proposes a two-layer scheme, where they use recurrent neural network (RNN) or CNN for sentence embedding, and another RNN for document embedding. Our method follows the two-layer scheme, and shares similar spirit with [Yang *et al.*, 2016] in using context vectors to distinguish informative words or sentences from non-informative ones, but instead of global context vectors, they can be dynamically generated for each datum when extra guidance is available. We also differs from these works in using extracted relevant law articles to support charge classification, which require us to distinguish relevant articles from irrelevant ones, and further aggregate the information from two sources for classification.

As for multi-label document classification, two loss functions are commonly used. The first one is binary cross entropy [Nam *et al.*, 2014], which treats the multi-label classification task as multiple binary classification tasks. The second one is cross entropy [Kurata *et al.*, 2016], which converts the multi-label target to label distribution during training, and use a threshold selected via validation set to generate the final prediction during testing. In our pilot experiments, we find the latter one converges faster and performs better, so we will use the latter one in this paper.

In the thread of work on charge prediction, [Liu *et al.*, 2004; Liu and Hsieh, 2006] use KNN to classify charges of criminal cases in Taiwan. Except for the inferior scalability of the KNN method, the word-level or phrase-level features also do not provide enough information to distinguish very similar charges. [Lin *et al.*, 2012] propose to make deeper understanding of the case by identifying key factors manually designed regarding two similar charges. Their method also suffers from sclability issue since the human efforts requied for designing and annotating these charge specific factors. Our method, however, employs RNN and attention mechnism to make comphrehensive understanding of the case, and all the training data are automatically constructed based on public judgement docuemnts.

[Liu and Liao, 2005; Liu and Hsieh, 2006] also try to find the specific law articles that has been violated. However, they convert the multi-label problem to multi-class classification problem by only considering a fixed set of article combinations, which can not scale well since the number of possible combinations will grow expentially when a larger set of law articles are considered. [Liu *et al.*, 2015] instead designs a scalable way to find relevant law articles, by first use Support Vector Machine (SVM) for preliminary article classification, and then rerank the results by using the similarity between the words in facts and articles, and the correlations among law articles as indicators. We also utilizes SVM to extract top $k$ articles, but instead use RNN and attention mechanism to better understand texts and the correlation among articles.

Another related thread of work is to predict the overall outcome of a case. The target can be predicting whether the outcome will side with the plaintiff or defendant [Aletras *et al.*, 2016], or will the present court affirm or reverse the decision of a lower court [Katz *et al.*, 2016]. Our work mainly differs from them in that, instead of binary outcome, we step further to focuse on the detailed results of the case, i.e., the charges, where the output may contain multiple labels.

Our work also shares similar spirit with the legal question answering task [Kim *et al.*, 2014a], which aims at answering the yes/no questions in Japanese legal bar exams, that we all believe that relevant law articles are important for decisions in civil law system. The task first requires participants to extract relevant Japanese Civil Code articles, and then these articles are used to answer the yes/no question. The article extraction phase is often treated as an information retrieval task, and the question answering phase is usually considered as a textual entailment task [Kim *et al.*, 2014b; 2015]

## 3 Data Preparation

Our data are collected from China Judgements Online[1]. The Chinese government has been publishing judgement documents on it since 2013. We randomly choose 50,000 judge-

[1] http://wenshu.court.gov.cn

…经审理查明，2011年10月6日凌晨，被告人AA携带改锥、扳手、破坏钳、刀等物品到尉氏县张市镇尹庄村BB家门口盗窃农用车上的电瓶时被被害人BB发现，在逃跑过程中AA为抗拒抓捕持刀将BB致伤。……

After hearing, our court identified that the defendant AA got spotted by the victim BB when he was trying to steel the battery of an agricultural vehicle on the morning of October 6th, 2011. AA wounded BB with a knife while BB was trying to catch him. ……

**Facts**

本院认为，被告人AA在盗窃过程中携带凶器，为抗拒抓捕而当场使用暴力致被害人BB轻微伤，其行为已构成抢劫罪，……依照《中华人民共和国刑法》第二百六十三条、第二百六十九条、…之规定，判决如下：

Our court hold that, the defendant AA caused BB minor wound during the process of stealing. His acts constituted the crime of robbery. …… According to the **Article 263**, **Article 269**, … of **the Criminal Law of the People's Republic of China**, the sentence is as follows:

**Court View**

被告人AA犯抢劫罪，判处有期徒三年，并处罚金人民币一千元。…

AA committed the crime of **robbery**, and shall be sentenced to a fixed-term imprisonment of 3 years and a fine of 1000 yuan. …

**Sentence**

Figure 1: Example judgement document excerpt of a Chinese criminal case. Names are anonymized as AA and BB.

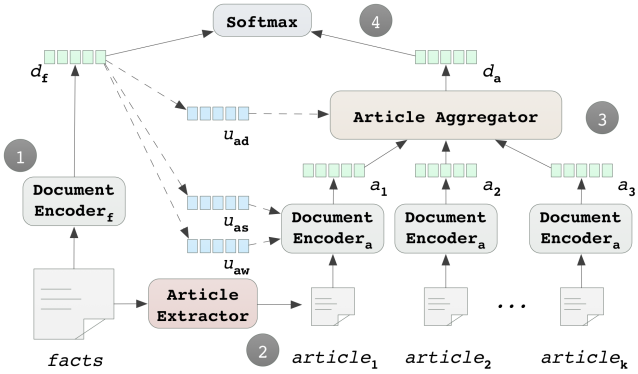


Figure 2: Overview of Our Model

ments as training data, 5,000 for validation and 5,000 for testing. To ensure enough training data for each charge, we only keep the charges that appear more than 80 times in the training data and other charges are used as negative data. As for law articles, we only consider the ones in the Criminal Law of the People's Republic of China. The resulting dataset contains 50 distinct charges and 321 distinct articles. About 3.56% cases contain more than one charges, and 94.18% cases contain more than one law articles. The facts part of each judgement contains 382.60 words on average.

An example judgement is shown in Figure 1 (starting from the facts part). Following the words descripting procedures like when the defendant is prosecuted, the fact description part often starts with the clause 经审理查明 (after hearing, our court identified that), the court view part often starts with the clause 本院认为 (our court hold that), and the final sentence part often starts with the clause 判决如下 (the sentence is as follows). We therefore divide the judgement document into different parts according to these indicator clauses. The articles are extracted in the court view part by regular expressions, and the charges are identified in the sentence part using string matching with a manually built charge list.

We only retain the cases with one defendant. Since when multiple defendants exist, it is hard to separately relate each defendant to his (or her) corresponding facts, articles and charges due to the unstructured nature of the judgement.

## 4 Our Approach

Our approach follows four steps as depicted in Figure 2. (1) The input fact description is fed to a document encoder to generate the fact embedding $\mathbf{d_f}$. (2) Concurrently, the input fact description is also passed to a relevant article extractor to find top $k$ relevant law articles. (3) The article are fed to another document encoder, and the article embeddings are further passed to an article aggregator to produce the aggregated article embedding $\mathbf{d_a}$. Meanwhile, three global context vectors, i.e., $\mathbf{u_{aw}}$, $\mathbf{u_{as}}$ and $\mathbf{u_{ad}}$, are generated from $\mathbf{d_f}$ for the article document encoder and the article aggregator. (4) Finally, $\mathbf{d_f}$ and $\mathbf{d_a}$ are concatenated and passed to a softmax classifier to predict the charge distribution of the input case.

### 4.1 Document Encoder

Intuitively, a sentence is a sequence of words and a document is a sequence of sentences. As suggested by previous works [Tang *et al.*, 2015; Yang *et al.*, 2016], the document embedding problem can be converted to two sequence embedding problems. As shown in Figure 4, we can first embedding each sentence using the sentence-level sequence encoder, and then aggregate them with document-level sequence encoder to generate the document embedding $\mathbf{d}$. Although it is possible to use different models for these two sequence encoders, we simply use the same architecture here for simplicity.

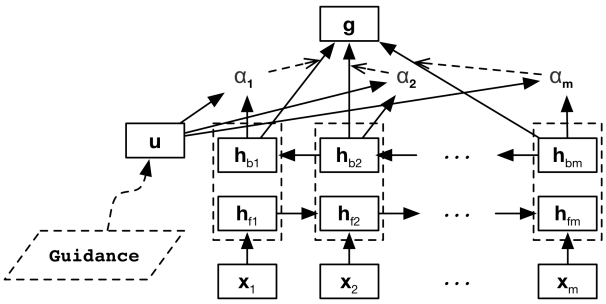


Figure 3: Attentive Sequence Encoder

**Bi-GRU Sequence Encoder** A prominent challenge in building a sequence encoder is how to take the correlation of each elements into consideration. A promising solution is

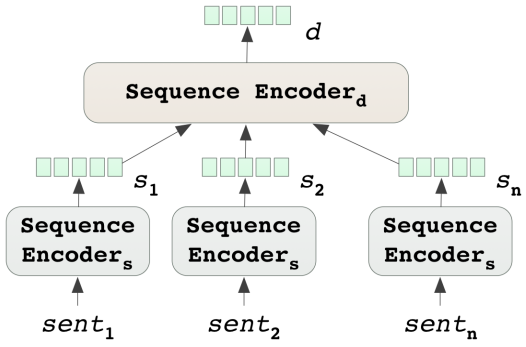


Figure 4: Document Encoder

the bi-directional gated recurrent unit (Bi-GRU) model proposed by [Bahdanau *et al.*, 2015], which encodes the context of each element by using a gating mechanism to track the state of sequence. Specifically, Bi-GRU first uses a forward and a backward GRU [Cho *et al.*, 2014], which a kind of Recurrent Neural Network (RNN), to encode the sequence in two opposite directions, and then concatenate the results of both GRUs to form the final outputs.

Given a sequence $[\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T]$ where $\mathbf{x}_t$ is the input embedding of element $t$, the state of Bi-GRU at position $t$ is:

$$\mathbf{h}_t = [\mathbf{h}_{ft}, \mathbf{h}_{bt}] \tag{1}$$

where $\mathbf{h}_{ft}$ and $\mathbf{h}_{bt}$ are the state the forward and backward GRU at position $t$ respectively. The final sequence embedding is either the concatenation of $\mathbf{h}_{fT}$ and $\mathbf{h}_{b1}$, or simply the average of the average of $\mathbf{h}_t$.

**Attentive Sequence Encoder** However, $[\mathbf{h}_{fT}, \mathbf{h}_{b1}]$ often fails to capture all the information when the sequence is long, and using the average of $\mathbf{h}_t$ also has the drawback that it treats useless elements equally with informative ones.

Inspired by [Yang *et al.*, 2016], we also use a context vector to attentively aggregate the elements, but instead of using a global context vector, we further allow it to be dynamically generated when extra guidance is available (see Section 4.2).

The framework of our attentive sequence encoder is show in Figure 3. Given the Bi-GRU state sequence $[\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_T]$, we calculate a sequence of attention values $[\alpha_1, \alpha_2, ..., \alpha_T]$ where $\alpha_t \in [0, 1]$ and $\sum_t \alpha_t = 1$. The final embedding of the sequence $\mathbf{g}$ is then calculated by:

$$\mathbf{g} = \sum_{t=1}^{T} \alpha_t \mathbf{h}_t \tag{2}$$

$$\alpha_t = \frac{exp(\mathbf{v}_t^T \mathbf{u})}{\sum_t exp(\mathbf{v}_t^T \mathbf{u})} \tag{3}$$

$$\mathbf{v}_t = tanh(\mathbf{W}\mathbf{h}_t) \tag{4}$$

where $\mathbf{W}$ is a weight matrix, and $\mathbf{u}$ is the context vector to distinguish informative elements from non-informative ones.

## 4.2 Using Law Articles

One of the difficulties of using law articles to support our charge prediction lies in the fact that statutory laws contain a large number of law articles, which makes complex classification models time-consuming in training, and run slowly in production environment as well.

We use a two-step approach to resolve this problem. Specifically, at the first step, we use a fast and easy-to-scale classifier to filter out a large fraction of irrelevant articles, and retain the top $k$ articles for the next step. Then, we use neural network to make conprehensive understanding of the top $k$ articles, and then use attention mechanism to select the most relevant ones for charge classification.

**Top $k$ Article Extractor** We consider the relevant article extraction task as multiple binary classification tasks. Specifically, since 321 distinct law articles in the Criminal Law of the People's Republic of China are mentioned in our dataset, we therefore build 321 binary classifiers where each classifier focuses on the relevance of a specific article. When more articles are considered, we can simply add more binary classifiers accordingly, with the existing classifiers untouched.

We use bag-of-words-based SVM as our binary classifier, which is fast and performs well in text classification [Joachims, 2002; Wang and Manning, 2012]. Specifically, we use bag-of-words TF-IDF features, chi-square for feature selection and SVM with linear kernel for binary classification. The articles are ranked by the output score and the top $k$ articles are retained for each judgement document.

**Article Encoder** Since each law article may contain multiple sentences, as shown in Figure 2, we also use the document encoder described in Section 4.1 to generate the embedding $\mathbf{a}_j, j \in [1, k]$ for the top $k$ articles. While using similar architecure, this document encoder differs from the one used for fact embedding that, instead of using global context vectors, the word-level and sentence-level context vectors, i.e., $\mathbf{u_{aw}}$ and $\mathbf{u_{as}}$, which are used to produce word-level and sentence-level attention values respectively, are generated dynamically for each case from the fact embedding $d$:

$$\mathbf{u}_{aw} = \mathbf{W}_w \mathbf{d} + \mathbf{b}_w; \quad \mathbf{u}_{as} = \mathbf{W}_s \mathbf{d} + \mathbf{b}_s \tag{5}$$

where $\mathbf{W}$ is the weight matrix and $\mathbf{b}$ is the bias. The dynamic context vectors enables us to attend to informative words or sentences with respect to the facts of each case, rather than just selecting generally informative ones.

**Attentive Article Aggregator** The article aggregator aims to generate an aggregated embedding for the top $k$ articles. Since not all the articles are relevant, the difficulty here is how to distinguish relevant articles from irrelevant ones.

Due to the inferior classification ability of the relevant article extractor, the order of the top $k$ articles are not very meaningful. Therefore, the top $k$ articles are more of a set than a sequence. However, as suggested by [Vinyals *et al.*, 2016], when encoding a set, it is still beneficial to embed the context of each element with a bidirectional RNN. In our task,

specifically, using bi-directional RNN is beneficial to model the co-existence tendency of relevant articles.

Therefore, we use the attentive sequence encoder in Section 4.1 to generate the aggregated article embedding $\mathbf{d}_a$. Again, to guide the attention with fact descriptions, we also dynamicall generate the article-level context vector $\mathbf{u}_{ad}$ by:

$$\mathbf{u}_{ad} = \mathbf{W}_d\mathbf{d} + \mathbf{b}_d \quad (6)$$

### 4.3 Output and Loss Function

To generate the charge prediction, we first concatenate the document embedding $\mathbf{d}_f$ and the aggregated article embedding $\mathbf{d}_a$, and feed them to two consecutive full connection layers to generate a new vector $\mathbf{d}$. Then, $\mathbf{d}$ is passed to a softmax classifier to generate the predicted charge distribution. We use the validation set to determine a threshold $\tau$, and consider all the charges with a output probability higher than $\tau$ as positive predictions. Also note that the input to the first full connection layer can also be $\mathbf{d}_f$ only, which means we do not use the information from relevant law articles.

As for training, we use cross entropy as our loss function:

$$Loss = -\sum_{i=1}^{N}\sum_{l=1}^{L} y_{il} log(o_{il}) \quad (7)$$

where $N$ is the number of training data, $L$ is the number of charges, $y_{il}$ and $o_{il}$ are the target and predicted probability of charge $l$ of datum $i$. Here the target charge distribution $\mathbf{y}$ is generated by setting positive labels to $\frac{1}{m}$ and negative labels to 0, where $m$ is the number of positive labels.

**Guided Article Attention** Since the judgement documents also contain gold standard relevant law articles, we can further use this information to guide the article attention during training. Specifically, given the top $k$ articles, we want the article attention distribution $\boldsymbol{\alpha} \in \mathbb{R}^k$ to simulate the target distribution $\mathbf{t} \in \mathbb{R}^k$, where $t_j = \frac{1}{k'}$ if article $j$ belongs to the gold standard articles and $t_j = 0$ otherwise. Here $k'$ is the number of gold standard articles in the top $k$ extractions.

We still use cross entropy here, and the loss function is:

$$Loss = -\sum_{i=1}^{N}(\sum_{l=1}^{L} y_{il} log(o_{il}) + \beta \sum_{j=1}^{k} t_{ij} log(\alpha_{ij})) \quad (8)$$

where $\beta$ is the weight for article attention guidance loss.

## 5 Experiments

### 5.1 Implementation Details

We use HanLP[2] for Chinese word segmentation and POS tagging. As for word embeddings, we use Baidu Encyclopedia, 3 million judgement documents and 3 million legal question answer pairs crawled from multiple legal forums as corpus, and the word2vec [Mikolov *et al.*, 2013] for training. The size of the resultant word embedding is 100-d and there are 573,353 words in total. We also randomly initialize a 50-d vector for each POS tag, which is concatenated with the word embedding to generate the final input. Each GRU in the Bi-GRU is of size 75, the two full connection layers are of size 200 and 150. The relevant article extractor generates top 20 articles, and the weight of the article attention loss is 0.1. We use Stochastic Gradient Descent (SGD) for training, with learning rate 0.1, and batch size 8. The chi-square feature selector keeps the top 2,000 features.

[2] https://github.com/hankcs/HanLP

| Model | Precision | Recall | F1 |
|---|---|---|---|
| *SVM* | **93.94**/79.53 | 77.66/49.54 | 85.03/61.05 |
| *SVM_article* | 91.77/71.33 | 72.10/45.85 | 80.76/55.82 |
| *NN* | 91.30/**83.32** | 87.39/74.99 | 89.31/78.94 |
| *NN_article* | 90.79/83.07 | 88.42/75.73 | 89.59/79.23 |
| ***NN_guide_article*** | 91.80/82.44 | **88.67/78.62** | **90.21/80.48** |
| *SVM_gold_article** | **98.97**/94.58 | 95.39/83.21 | 97.15/88.53 |
| ***NN_gold_article**** | 98.78/**95.26** | **98.24/95.57** | **98.51/95.42** |
| *NN_article_only* | 87.38/78.61 | 82.69/64.32 | 84.97/70.75 |
| *NN_gold_only** | 97.22/92.39 | 98.36/94.73 | 97.79/93.55 |

Table 1: Charge Prediction Results. * means we use gold standard relevant law articles, which are not available in production environment.

### 5.2 Charge Prediction Results

We compare our method with a baseline BOW-based SVM method, which is similar to the one used for the top $k$ article extractor, except that the outputs are charges rather than articles. The results are summarized in Table 1. The left side of the slash refers to micro statistics, and the right side refers to macro statistics. Specifically, micro precision (or recall) is the the number of correct predictions divided by the total number of predictions (or the total number of gold standard charges), while the macro precision (or recall) is the sum of the precision (or recall) of each charge divided by the number of distinct charges in test set. The micro (or macro) F1 is the harmonic mean of the micro (or macro) precision and recall.

We can see that, the basic `SVM` model, which only use facts as input, proves to be a strong baseline. If we use both the facts and the gold standard articles of each case (`SVM_gold_article`), the performance can be further improved significantly, showing that the relevant law articles contain valuable information for charge prediction. However, if we replace the gold standard articles with the extracted ones (`SVM_article`), the results become rather worse, indicating that the SVM model cannot benefit from the additional information of relevant articles when they are noisy.

On the other hand, since our attentive article aggregator has the ability to distinguish relevant articles from extraction mistakes, our neural network (NN) model (`NN_article`) can make use of the noisy article extractions, and improve the performance over the model using only facts as input (`NN`), which has already ourperfroms `SVM` by a large margin. Furthermore, if we use the gold standard articles to guide the article attention during training (`NN_guided_article`), the performance can be further improved. Similar to SVM models, if we use gold standard articles (`NN_gold_article`) instead of extracted ones, there will exist a clear improvement as well, and without surprise, it also outperforms `SVM_gold_article`. Also note that, compared with the

| | Top_5 | Top_10 | Top_20 | Top_30 |
|---|---|---|---|---|
| *Recall* | 77.60 | 88.96 | 94.21 | 96.53 |
| *NDCG* | 80.28 | 84.32 | 86.47 | 87.24 |

Table 2: Top $k$ Article Extraction Performance

| $\beta$ | Prec@1 | Prec_Full | MAP | Charge_F1 |
|---|---|---|---|---|
| *0* | 60.94 | 50.76 | 61.61 | 89.59/79.23 |
| *0.01* | 81.06 | 68.61 | 78.00 | 89.77/79.48 |
| *0.1* | 87.90 | 74.14 | 83.39 | **90.21/80.48** |
| *1* | **92.66** | **80.43** | **88.24** | 89.83/78.66 |

Table 3: Article Attention Performance

corresponding SVM models, the improvements made by our NN architecture are most prominent in macro statistics, showing that the the SVM model has a strong bias towards frequent charges, while the NN models have a more balanced performance between frequent and infrequent ones. This is probably due to the nature of the BOW feature, which is unable to handle synonyms. However, since our NN model takes pre-trained word embeddings as input, it can better understand the meaning of a word even when it is rare in the training data.

We also evaluate our NN model with only articles as input. Specifically, the relevant articles are concatenated and passed to the model by replacing the fact descriptions. We find that using only the top 20 extracted articles (`NN_article_only`) still generates fair results, and the model performs very good when the extracted articles are replaced with gold standard ones (`NN_gold_only`), which significantly outperforms `NN`. This actually indicates the nature of civil law system that judgements are made based on statutory laws rather than decisions of previous cases. However, we can see that the performance of `NN` is also promising, showing that although the judgments are based on statutory laws, our model can still learn the legal logic behind this implicitly through only the (facts, charges) paris. Also note that `NN_gold_only` is not as good as `NN_gold_article`, showing that the facts can provide additional information for charge prediction even if we use gold standard law articles.

### 5.3 Article Related Results

**Top $k$ Article Extraction** The performance of our top $k$ article extractor is show in Table 2. We can see that, although simple, our SVM article extractor already achieves promising performance, and the recall of the top 20 results has achieved 94.21%, which is good enough to support charge prediction. On the other hand, however, we still cannot trust its prediction results directly. In our test set, the micro F1 of the model is only 61.08%, which will lead to severe error propagation problem if we use the precition directly. Therefore, we instead retain the top 20 articles, and use the powerful NN model to resolve the noise in the top 20 results.

**Article Attention** We can consider the article attention module as a re-ranking function over the extracted top $k$ articles, and accordingly use the gold standard articles in the $k$ articles to evaluate the re-ranking performance. Table 3 shows the model performance on article attention (column 2-4) and charge prediction (colum 5), under different article attention loss weights ($\beta$ in Equation 8). `Prec@1` refers to top 1 precision, and `Prec_Full` refers to full precision. For example, if there are $k'$ gold standard articles in the retained $k$ articles, then `Prec_Full` refers to the precision of the top $k'$ articles.

We can see that, even if there is no guidance over the article attention ($\beta = 0$), our model still has reasonable performance on re-ranking the $k$ articles. When attention guidance is added, the attention quality improves significantly, and it keeps increasing as $\beta$ goes up. The fair performance of the article attention indicates that our model can provide reasonable legal basis to support the charge prediction. However, the charge prediction performance does not always increase with the article attention quality, and the best result is achieved when $\beta = 0.1$. This shows that there exists a tradeoff between the benefits of more accurate article attention and the less model capacity left for charge classification due to the increased emphasis on the article attention performance.

| Model | Precision | Recall | F1 |
|---|---|---|---|
| *SVM* | **100.00**/56.00 | 40.20/34.64 | 57.34/42.80 |
| *NN* | 87.14/71.30 | 59.80/54.92 | 70.93/62.05 |
| ***NN_article*** | 87.18/**75.24** | 66.67/**60.37** | 75.56/**66.99** |
| ***NN_guide_article*** | 90.00/68.72 | **70.59**/57.50 | **79.12**/62.61 |

Table 4: Results on News Data

### 5.4 Performance on News Data

Since there are some differences between the words used by laymen and legal practitioners, we manually annotated 100 news[3], which contains an average of 261.79 words and 25 distinct charges, to see how the model trained on judgement documents performs on the fact descriptions written by ordinary people. The results are shown in Table 4.

We can see that, although the SVM model has good performance on the judgement document data, it suffers from a significant performance drop on news data. Recall that the SVM model is based on BOW, this performance drop actually indicates an innegligible word usage gap between laymen and legal practitioners. However, the high micro precision of `SVM` indicates that, although cannot generalizes well, the patterns learned by the SVM model are very reliable in themselves. As for our NN models, although there also exists a performance drop, they performs much better than the SVM model. This shows that the word usage gap can be resolved by using word embeddings to a some extent. Furthermore, we can see that the model using relevant articles (`_article`) clearly outperform `NN`, showing that relevant articles provide valuable information in this situation.

Also note that, since the news test set is relatively small regarding the number of charges, the macro statistics are not as meaningful as those in the judgement document test set. For example, correctly predicting a charge with only one instance will improve the macro precision and recall by 4% ($1 \div 25$). Therefore, `NN_article` outperforms

[3]the news is randomly selected from `http://www.news.cn/legal/` and `http://legal.people.com.cn/`

NN_guide_article in macro F1 by 4.38% does not necessarily mean that it performs better on infrequent charges. Actually, NN_article only generates one more such case than NN_guide_article, while NN_guide_article correctly predicts 4 more cases than NN_article in total. Therefore, we still consider NN_guide_article to be better than NN_article on news data.

## 6 Conclusion

In this paper, we proposes that the public judgement documents are natural high quality training data for fact based charge prediction. Since China uses statutory law, it is reasonable to assume that finding relevant articles that can be applied to the case can help the charge prediction task. TODO: elaborate on this Based on this assumption, we propose a novel neural network model to use both the fact and the relevant articles for charge prediction. The experiments show that our model performs better than the bag-of-words baseline model as well as the neural network model that only uses fact descriptions. We also shows that the neural network model trained on the judgement document data can achieve reasonable performance on news data.

## References


[Aletras *et al.*, 2016] Nikolaos Aletras, Dimitrios Tsarapatsanis, Daniel Preoţiuc-Pietro, and Vasileios Lampos. Predicting judicial decisions of the european court of human rights: A natural language processing perspective. *PeerJ Computer Science*, 2:e93, 2016.

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of ICLR*, 2015.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[Joachims, 1998] Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer, 1998.

[Joachims, 2002] Thorsten Joachims. *Learning to classify text using support vector machines: Methods, theory and algorithms*. Kluwer Academic Publishers, 2002.

[Katz *et al.*, 2016] Daniel Martin Katz, II Bommarito, J Michael, and Josh Blackman. A general approach for predicting the behavior of the supreme court of the united states. *arXiv preprint arXiv:1612.03473*, 2016.

[Kim *et al.*, 2014a] Mi-Young Kim, Randy Goebe, and Ken Satoh. COLIEE-14. `http://webdocs.cs.ualberta.ca/~miyoung2/jurisin_task/index.html`, 2014.

[Kim *et al.*, 2014b] Mi-Young Kim, Ying Xu, and Randy Goebel. Legal question answering using ranking svm and syntactic/semantic similarity. In *JSAI International Symposium on Artificial Intelligence*, pages 244–258. Springer, 2014.

[Kim *et al.*, 2015] Mi-Young Kim, Ying Xu, and Randy Goebel. A convolutional neural network in legal question answering. *In Proc. JURISIN*, 2015.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[Kurata *et al.*, 2016] Gakuto Kurata, Bing Xiang, and Bowen Zhou. Improved neural network-based multi-label classification with better initialization leveraging label co-occurrence. In *Proceedings of NAACL-HLT*, pages 521–526, 2016.

[Lin *et al.*, 2012] Wan-Chen Lin, Tsung-Ting Kuo, and Tung-Jia Chang. Exploiting machine learning models for chinese legal documents labeling, case classification, and sentencing prediction. *ROCLING XXIV (2012)*, page 140, 2012.

[Liu and Hsieh, 2006] Chao-Lin Liu and Chwen-Dar Hsieh. Exploring phrase-based classification of judicial documents for criminal charges in chinese. In *International Symposium on Methodologies for Intelligent Systems*, pages 681–690. Springer, 2006.

[Liu and Liao, 2005] Chao-Lin Liu and Ting-Ming Liao. Classifying criminal charges in chinese for web-based legal services. In *Asia-Pacific Web Conference*, pages 64–75. Springer, 2005.

[Liu *et al.*, 2004] Chao-Lin Liu, Cheng-Tsung Chang, and Jim-How Ho. Case instance generation and refinement for case-based criminal summary judgments in chinese. *Journal of Information Science and Engineering*, 20(4):783–800, 2004.

[Liu *et al.*, 2015] Yi-Hung Liu, Yen-Liang Chen, and Wu-Liang Ho. Predicting associated statutes for legal problems. *Information Processing & Management*, 51(1):194–211, 2015.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[Nam *et al.*, 2014] Jinseok Nam, Jungi Kim, Eneldo Loza Mencía, Iryna Gurevych, and Johannes Fürnkranz. Large-scale multi-label text classification—revisiting neural networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 437–452. Springer, 2014.

[Tang *et al.*, 2015] Duyu Tang, Bing Qin, and Ting Liu. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1432, 2015.

[Vinyals *et al.*, 2016] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638, 2016.

[Wang and Manning, 2012] Sida Wang and Christopher D Manning. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 90–94. Association for Computational Linguistics, 2012.

[Yang *et al.*, 2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016.