IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON ADVANCES IN MACHINE LEARNING AND COGNITIVE COMPUTING FOR INDUSTRY APPLICATIONS

Received April 4, 2020, accepted April 12, 2020, date of publication April 17, 2020, date of current version May 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2988493

Few-Shot Learning for Chinese Legal Controversial Issues Classification

YIN FANG[1], XIN TIAN[1], HAO WU[1], SONGYUAN GU[2], ZHU WANG[2], FENG WANG[3], JUNLIANG LI[3], AND YANG WENG[1]

[1]College of Mathematics, Sichuan University, Chengdu 610065, China
[2]Law School, Sichuan University, Chengdu 610207, China
[3]Union Big Data Technology, Chengdu 610041, China

Corresponding author: Yang Weng (wengyang@scu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC0830300.

ABSTRACT Chinese courts organize debates surrounding controversial issues along with the gradual formation of the new procedural system. With the progress of China's judicial reform, more than 80 million judgement documents have been made public online. Similar controversial issues identified in and among the massive public judgment documents are of significant value for judges in their trial work. Hence, homogeneous controversial issues classification becomes the basis for similar cases retrieval. However, controversial issues follow the power-law distribution, not all of them are within the labels provided by manual annotation and their categories cannot be exhausted. In order to generalize those unfamiliar categories without necessitating extensive retraining, we propose a controversial issues classification algorithm based on few-shot learning. Two few-shot learning algorithms are proposed for our controversial issues problem, Relation Network and Induction Network, respectively. With only a handful of given instances, both of them have shown excellent results on the two datasets, which proves their effectiveness in adapting to accommodating new categories not seen in training. The proposed method provides trial assistance for judges, promotes the dissemination of experience and improves fairness of adjudication.

INDEX TERMS Controversial issues, few-shot learning, text classification, power-law, BERT.

I. INTRODUCTION

As China is making continuous progress in social development, its judicial reform is bound to advance further. The reform proceeds from the demands of the public for justice, with strengthening supervision and restraint of power as priority. As a matter of fact, judicial openness has been thought of as a key factor in safeguarding the public right to know, participate, and supervise. With the high-speed development of information technology, issuance of judgements on the internet plays a pivotal role in the promotion of judicial openness.

Along with the gradual formation of the new procedural system, Chinese courts organize debates surrounding controversial issues [1]. Controversial issues are at the core of conflicts between the parties. ''Did the defendant infringe the plaintiff's portrait right?'' ''Whether the plaintiff is a competent subject?'' ''Is the amount of compensation claimed by the plaintiff reasonable?'' The above are examples of controversial issues in different disputes. In order to ascertain the facts and then carry out legal reasoning, the judges divide controversial issues into the factual controversial issues and legal controversial issues. The factual controversial issues contribute to focusing facts investigation in court adjudication, while the legal controversial issues are helpful to court debate organization and legal application. Both of them have been thought of as essential elements in enhancing court efficiency. The written judgement shows the process of the legal argumentation, which contains controversial issues that have been collated, investigated, and debated during the court trials. Therefore, controversial issues play key roles in restoration of trial scene and judges' decision making.

In particular, we divide controversial issues in judgements into four categories. In controversial issue group of repeated cause of action (G1), at the request of both parties, the judges consider that controversial issues are actually the causes of the action involved in the cases. Cause of the action represents the summary of the characters and content of the specific lawsuit. Controversial issue group of general procedure law (G2) concludes procedural controversial issues that may exist in

The associate editor coordinating the review of this manuscript and approving it for publication was Min Xia.

different causes of action. The characteristic of controversial issue group of general substantive laws (G3) is that the judges need to make value judgement on whether minor premise (facts of a case) meets major premise (legal provisions) according to the explicit provisions of law. Non-general legal controversial issue and factual controversial issue group (G4) concludes controversial issues in related to the facts of the cases, which have great reference significance. However, G3 and G4 account for the majority of them. It indicates that most of controversial issues are closely related to the facts and legal provisions of the causes of the action to which they belong. As there are controversial differences between different causes of the action, it is necessary to study separately. It not only realizes legal knowledge construction of controversial issues under various causes of the action, but also provides convenience for judges to retrieve controversial issues.

To date, the number of judgements published online is more than 80 million. However, efficiently identifying critical information in massive data will be a tremendous challenge. Another common phenomenon we must not overlook is that, due to the limited number of cases that individual judge has access to, it is difficult for them to draw on experiences of other judges in summarizing controversial issues and conducting trials, which has greatly hindered the dissemination of experience and the accumulation of legal knowledge. In particular, unlike controversial issues with formatted expressions in the other three categories, judges' descriptions are always irregular in G4. Therefore, it is necessary to classify cases with different types of controversial issues in G4. In the circumstances, homogeneous controversial issues classification becomes the basis for cases classification. Homogeneous controversial issues refer to controversial issues with different expressions but have the same substantial meaning at the legal level. When judges face difficulties in making decisions, they can draw lessons from how other cases with homogeneous controversial issues were judged. At the same time, they can refer to the format of other judges in summarizing controversial issues, which makes the expression more standardized. However, due to the huge corpus, different expressions and numerous categories, it is expensive to manually distinguish homogeneous controversial issues. Machine learning algorithms are suitable for solving this problem of heavy workload and inefficiency.

Owing to the administrative judges' discretion and extra-judicial factors, along with the uncertainty of facts and legal, judges' descriptions are unformatted. Because of the complexity of human language expression, ambiguity arises in the classification of controversial issues. Here we give two examples of product liability disputes. Two sentences like ''Whether compensation should be paid for vehicle depreciation loss due to engine damage?'' ''Whether to compensate the difference between pre-sale and after-sale due to vehicle engine damage?'', with different expressions but the same semantics, may be divided into distinct categories. The same situation happens in ''Whether punitive damages can be supported?'' ''Can triple compensation be supported?'' and ''Can ten times compensation be supported?''. In product liability disputes, different multiples of compensation are caused by different multiples of punitive damages stipulated by law. Therefore, these three sentences belong to the same category legally. Classifying controversial issues accurately becomes a challenging natural language processing (NLP) task.

A surge of work proposed representing words as dense vectors, most of which are derived using various training methods inspired from neural-network language modeling. These representations, referred to as ''word embeddings'', have been shown to perform well in a variety of NLP tasks. Previous studies converted words to low dimensional vectors with Word2vec [2] in order to obtain the semantic information in controversial issues. However, Word2vec is a static word embedding model. Since the word vectors it trains are fixed, the problem of polysemy cannot be solved, and some domain-specific technical terms tend to be misunderstood. To improve word embeddings and better capture the deep semantic information of legal text, we utilize fastText [3] to compute word representations. FastText enables to compute embeddings for words which do not appear in the training data and train models on large corpora quickly. Further, we apply Chinese BERT [4] to obtain character representations. BERT uses two unsupervised prediction tasks to pre-train deep bidirectional representations and modifies the input and output of models in fine-tuning.

By studying the structure of datasets, we find evidence that controversial issues follow the power-law distribution [5], a small number of classes gather at the top of the distribution and take up the great majority of the whole controversial issues. Since such social problems are mostly composed of complex networks that follow the power-law distribution [6], [7], it is not surprising that the data structure of controversial issues has similar properties. The power-law distribution of controversial issues indicates that a few classes of them are common, while most classes are rare. Therefore, controversial issues are assigned into many classes, and class imbalance caused by the huge differences in the number of controversial issues contained in each class lead to a performance degradation of text clustering. So we apply supervised learning to deal with this problem. Text classification technology has been successfully used to achieve state-of-the-art performance in a variety of applications such as spam recognition [8], sentiment analysis [9] and public opinion monitoring [10]. But most of these algorithms often malfunction when forced to make predictions about data for which little supervised information is available. Large quantity work of data annotation needs to be implemented.

The tasks of data annotation are to figure out how many classes of controversial issues there are in total, and what kinds of controversial issues each classes contains. In order to solve the problem of low annotation efficiency and get high quality labeled data, clustering algorithms and topic model are adopted. Previous study utilized the most prevalent

clustering method K-means [11] when processing legal text data [12]. Text clustering applies cluster analysis to text, which uses machine learning and NLP to understand and categorize unstructured, textual data. Clustering algorithm is defined as an unsupervised technique for discovering whether the individuals of a population fall into different groups by making quantitative comparisons of multiple characteristics [13]. After clustering phase, the vast majority of homogeneous controversial issues are properly merged into the same cluster. Due to the semantic complexity and unformatted of legal texts, there are still many controversial issues that fall into the wrong places. In this situation, manually remove or merge controversial issues and their clusters need to be applied. For each cluster, rather than manually identifying the main information of controversial issues, it is more ideal to use a few words, which we called cluster labels, to summarize the topic of this cluster. By automatically obtaining cluster labels with LDA [14], experts can quickly determine if there are uninvited guests in each cluster through cluster labels and search for clusters with similar cluster labels to find out whether they should be merged, which contributes to achieve effectively data annotation.

However, since the total number of judgements published online is more than 80 million and the number of cases under different causes of the action is also very large, it is not realistic to conduct data annotation on all classes of controversial issues. Driven by the desire of generalizing those unseen classes without necessitating extensive retraining, few-shot learning [15] emerges as the times require. Given only a handful of instances, few-shot classifier can adapted to accommodate new classes not seen in training [16], which fits our scenario well. As it extracts some transferable knowledge through meta-learning, few-shot learning avoid overfitting caused by applying traditional deep learning methods on sparse data tasks. Through few-shot learning, controversial issues can be classified with a limited amount of labeled data.

In this paper, we introduce a controversial issues classification algorithm based on few-shot learning. Since the number of cases in labour disputes is large, and the semantic diversity of product liability disputes is complex, we choose judgments of these two causes of the action to conduct our experiments. In pre-processing module, we obtain legal text embeddings from the most advanced Chinese BERT pre-training model and apply hierarchical agglomerative clustering(HAC) algorithms in controversial issues clustering. We show respectively on the judgments of labour disputes and product liability disputes that our method is largely superior to the latest results. In order to allow experts to quickly access cluster topics without having to read verbatim and figure out miscategoried controversial issues faster in data annotation, we label each cluster with the LDA topic model. Each label extracts the critical information of the cluster, which solves the problem of heavy workload and time-consuming in data annotation. In classification module, we utilized two few-shot learning algorithms, Relation Network and Induction Network. Given only a handful of instances, experiment results suggest that few-shot learning models significantly outperforms existing classification methods and improves the accuracy by 11.29% and 7.28% in the two datasets, which proves their effectiveness in adapting to accommodate new classes not seen in training.

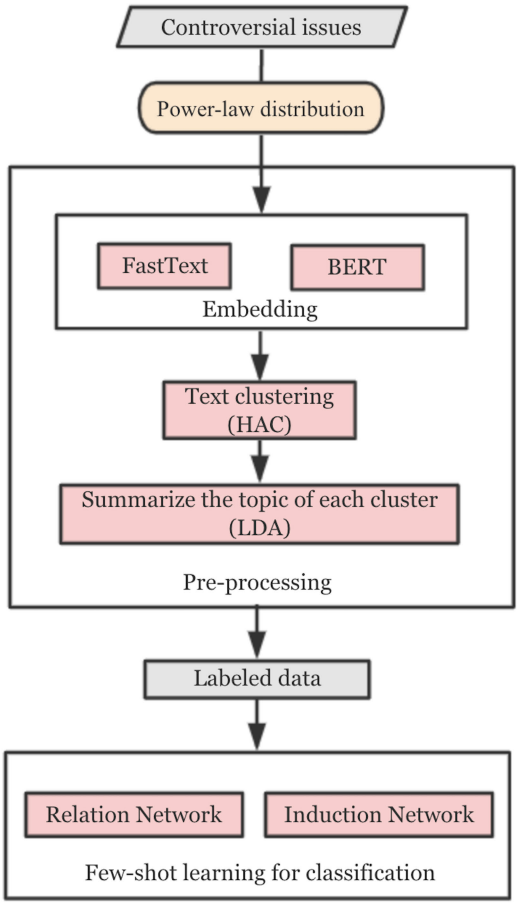


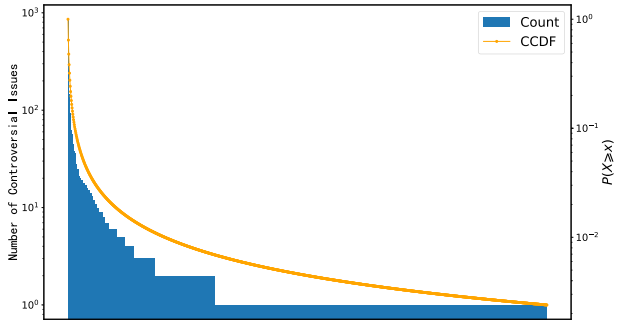
FIGURE 1. The overall flowchart of controversial issues classification.

It is also necessary to note that with the text clustering algorithms and cluster labeling, experts can conduct data annotation more convenient. And through few-shot controversial issues classification, judges can draw on experiences of other judges in summarizing controversial issues and conducting trials, which promotes the dissemination of experience and the accumulation of legal knowledge. Especially for complex cases, judges need to refer to cases with homogeneous controversial issues, which highlights the importance of industry applications of NLP and machine learning in providing trial assistance and improving the fairness of adjudication.

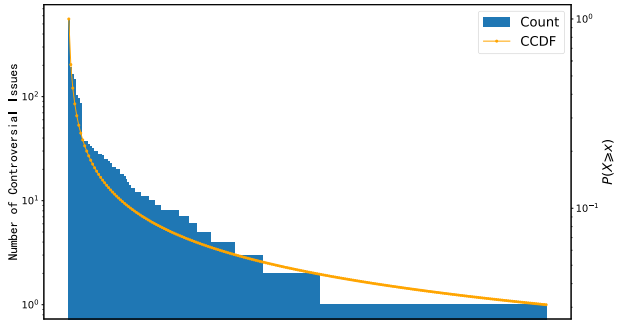
The rest of the paper is structured as follows. Data exploratory analysis is detailed in Section II. In Section III and IV we describe relevant algorithms. We present our experiments and results in Section V. In Section VI we conclude.

II. DATASET AND EXPLORATORY ANALYSIS

Since the number of cases in labour disputes is large, and the semantic diversity of product liability disputes is complex, we derived our dataset from the adjudicative documents in two causes of the action, labour disputes and product liability disputes. From 2014 to 2018, 605345 cases of labour disputes and 25827 cases of product liability disputes were published online. We randomly chose 5521 cases of labour disputes and 2570 cases of product liability disputes from them in G4 to conduct experiments and extracted controversial issues with regular expression.



(a) Labour Disputes



(b) Product Liability Disputes

FIGURE 2. The distribution and CCDF curve of controversial issues. The left *y*-axis represents the amount of controversial issues and the right *y*-axis denotes *P*(*x*). For blue histogram, *x*-axis describes 1295 and 241 different classes respectively for labour disputes and product liability. And for the yellow CCDF curve, *x*-axis represents random variable *x*.

In order to observe the structure of the datasets more intuitively, we describe it explicitly in pictures. In Figure 2a, we depicts the number of controversial issues in various classes as blue histogram, whose x-axis shows different 1295 classes while left y-axis represents the amount of controversial issues in labour disputes. The histogram describes the phenomenon that a small number of classes gather at the top of the distribution and take up the great majority of the whole controversial issues. In other words, it indicates that a few classes of controversial issues are common while most classes are rare. We find out that controversial issues follow the power-law distribution [5], with the ''long tail'' of clusters include negligible numbers of controversial issues. The power-law states that a relative change in one quantity causes a proportional change in another. Mathematically, a quantity x obeys a power law if it is drawn from this probability distribution:

$$p(x) \propto x^{-\alpha}. \tag{1}$$

Here, α is a constant parameter. The portion of the distribution having many classes of controversial issues far from the ''head'' of the distribution called the ''long tail''. In''long-tailed'' distributions, the events at the far end of the tail have a very low probability of occurrence, as it gradually ''tails off'' asymptotically.

We also compute the complementary cumulative distribution function (CCDF) of the power-law distributed variable, showed as yellow curve in Figure 2a. Here, x-axis represents the random variable x and y-axis is defined to be $P(x) = \Pr(X \geq x)$. $P(x)$, probability that the number of controversial issues in one class is greater than or equal to x, gradually decreases with the increase of x, which confirms our judgment that controversial issues follow the power-law distribution. Since we fitted the distribution to the number of samples in each class, the yellow curve is actually a line chart, which is the CCDF of the power-law distributed discrete variable.

Since social problems are mostly composed of complex networks that follow the power-law distribution, it is not surprising that the data structure of controversial issues has similar properties. Without loss of generality, we explore another cause of action named product liability disputes. We used the same setting with Figure 2a that the left y-axis represents the amount of controversial issues and the right one denotes $P(x)$. x-axis also has two meanings for different part of the figure. x-axis describes 241 different classes for blue histogram while for yellow curve, x-axis represents random variable x. Figure 2b demonstrates that similar situation occurs in product liability disputes, with a small portion of classes accounting for the vast majority of controversial issues.

According to our analysis, the ''long tail'' is formed by classes that include a negligible number of controversial issues, which is not representative. And a few classes of controversial issues occupy the vast majority. Therefore, in few-shot learning experiments, we only include classes with more than 6 samples, which is detailed in Section V-B.

III. PRE-PROCESSING

In this section we present pre-processing module, which contains word embeddings, text clustering and cluster labeling. More specifically, we first utilized Word2vec, fastText and BERT to capture the linguistic and semantic information of legal text. To figure out how many categories are in total and what types of controversial issues each category contains, we adopted both flat clustering and hierarchical clustering algorithms to initially merge homogeneous controversial issues into the same cluster. The semantic complexity and

unformatted format of legal text may cause controversial issues falling into the wrong clusters, or clusters with the same semantics not being merged into the same cluster. For each cluster, instead of identifying the main information of controversial issues manually, it is more ideal to summarize the topic of this cluster with a couple of words. We applied LDA to get cluster labels automatically, which solves the problems of heavy workload and inefficiency in data annotation.

A. EMBEDDING METHODS

To capture the semantic information in controversial issues and convert legal text to low dimensional vectors, several embedding methods were adopted.

Word embedding methods represent words as continuous vectors in a low dimensional space which capture lexical and semantic properties of words. The Word2vec model, which introduced by Tomas Mikolov *et al.* [2], has gained a lot of attention. The vector representations of words learned by Word2vec model have been proven to capture semantic meanings and are pivotal in various NLP tasks. Word2vec contains two architectures: continuous Bag-of-Words (CBOW) and continuous Skip-Gram. CBOW tends to predict the current word based on the contexts, while SkipGram tries to classify context words based on current words. Typically, we take the average of each term vectors as the meaning of a longer piece of text containing multiple terms.

FastText is a library created by Facebook's AI Research (FAIR) lab for efficient learning of word representations and sentence classification. FastText has gained a lot of attention in the NLP community as it has shown outstanding results in various NLP domains. Piotr Bojanowski proposed the model that can learn word representations while taking into account morphology [3]. They modeled morphology by considering subword units, and representing words by a sum of its character n-grams. FastText enables to compute representations for words that do not appear in the training data. Also, it is fast, allowing to train models on large corpora quickly.

BERT, which stands for Bidirectional Encoder Representations from Transformers, is a new language representation model proposed by Jacob Devlin *et al.* in 2018 [4]. BERT demonstrates new state-of-the-art performance on NLP tasks. In pre-training procedure, unlike Peters *et al.* (2018) [17] and Radford *et al.* (2018) [18], which uses traditional left-to-right or right-to-left language models for pre-training, BERT uses two unsupervised prediction tasks to pre-train deep bidirectional representations. They masked a part of the input tokens at random, and only those masked tokens are predicted. In addition, they pre-trained a binarized next sentence prediction task, in order to train a model that understands sentence relationships. For fine-tuning, the input and output of models in NLP tasks were simply modified, while parameters were learned during fine-tuning.

Google released the source code of BERT on Github. It also provides the BERT-Base and BERT-Large models pre-trained on Wikipedia with tensor processing unit, which includes a Chinese BERT pre-training model based on character level. Therefore, we can directly obtain semantic sentence embeddings which capture the linguistic and philosophical meaning.

B. CLUSTERING ALGORITHMS

The classic reference for clustering in pattern recognition, covering both K-means and EM, were proposed by Duda *et al.* in 2000 [19]. Rasmussen [20] introduced clustering in information retrieval (IR) field. The cluster hypothesis is due to Jardine and van Rijsbergen [21] who state it as follows: Associations between documents convey information about the relevance of documents to requests.

The Expectation-Maximization (EM) algorithm is a broadly applicable approach to the iterative computation of maximum likelihood estimates, useful in a variety of incomplete-data problems. In each iteration of the EM algorithm, there are two steps called the expectation step and the maximization step. This algorithm was introduced by Dempster, Laird, and Rubin [22] in their fundamental paper in 1977. K-means algorithm is a variant of EM, with the assumptions that clusters are spherical. K-means is widely used for clustering, compressing, and summarizing vector data. It has been identified as one of the top 10 algorithms in data mining. The most popular algorithm for K-means is known as Lloyd's algorithm [11].

Flat clustering is efficient and conceptually simple, but it also has several drawbacks. It requires a prespecified number of clusters as input and returns a flat, unstructured set of clusters. A heuristic method [23] that gets around this problem is to estimate $RSS_{min}(K)$ as follows. It performs i clusterings with a fixed K (each with a different initialization) and computes RSS of each. Then it takes the minimum of the i RSS values. After this procedure, we can inspect the values as K increases and find the "knee" in the curve.

In contrast, hierarchical clustering does not require us to prespecify the number of clusters. It outputs a hierarchy, a structure that is more informative than the unstructured set of clusters returned by flat clustering. Early references for specific hierarchical clustering algorithms are provided by King in 1967 (single-link) [24], Sneath *et al.* in 1973 (complete-link, GAAC) [25] and Lance *et al.* in 1967 (discussing a large variety of hierarchical clustering algorithms) [26]. There is evidence that hierarchical clustering tends to be more prominent, due to its very little premise with data characteristics and analysts' prior knowledge.

Hierarchical clustering algorithms are either top-down or bottom-up. Bottom-up algorithm is more frequently used in IR than top-down clustering, and it is the main subject of this section. Bottom-up hierarchical clustering, which is also called hierarchical agglomerative clustering or HAC, treat each document as a different cluster at the outset, and then successively agglomerate pairs of clusters until all clusters have been merged into a single cluster containing all documents.

Ward's method [27], also called the Ward variance minimization algorithm, is an important HAC technique which

select the merge with the smallest *RSS* in each step. Suppose there are $|u|$ original observations $u_0,\ldots, u_{|u|-1}$ in cluster u and $|v|$ original objects $v_0,\ldots, v_{|v|-1}$ in cluster v. Let v be any remaining cluster in the forest that is not u. The distance between the newly formed cluster u and each v is computed as follows,

$$d(u,v)=\sqrt{\frac{|v|+|s|}{T}d(v,s)^2+\frac{|v|+|t|}{T}d(v,t)^2-\frac{|v|}{T}d(s,t)^2} \tag{2}$$

where u is the newly joined cluster consisting of clusters s and t, v is an unused cluster in the forest, and $T = |v| + |s| + |t|$.

However, the hierarchy needs to be cut at some point in some cases. In our study, we impose a penalty for each new cluster. Conceptually, we start with a single cluster containing all documents and then search for the optimal number of clusters K by successively incrementing K by one:

$$K=\arg\min_{K'}[RSS(K')+\lambda K'] \tag{3}$$

where K' refers to the cut of the hierarchy that results in K' clusters, *RSS* is the residual sum of squares and λ is a penalty for each additional cluster. We can obtain a series of K with growing λ, and find the ''knee'' in the curve – the point where the successive decreases in K become noticeably smaller.

The main purpose of controversial issues clustering are to figure out how many categories they have in total, and what types of controversial issues each category contains. After clustering, we properly merged the majority of homogeneous controversial issues into the same cluster. Although the clustering results have already achieved good performance, owing to the semantic complexity and unformatted of legal texts, there are still some controversial issues that fall into the wrong clusters or clusters with the same semantics are not merged. Under these circumstances, individual controversial issues and their clusters need to be manually removed or merged.

C. CLUSTER LABELING

Rather than manually identifying the main information of controversial issues in each cluster, it is more ideal to use a few words to summarize the topic of this cluster. In this section, we propose to use LDA to automatically obtain these generalized words, which we called cluster labels. Experts can quickly determine if there are uninvited guests in each cluster through cluster labels and search for clusters with similar cluster labels to find out whether they should be merged. Cluster labeling mechanism avoids problems caused by time consuming and heavy workload when reading each controversial issues verbatim, which makes it more convenient to conduct data annotation.

LDA [14] is a topic model that has gained popularity among theoreticians and practitioners, and it serves as a tool for automatic corpus summarization and visualization. Processing fully generative semantics, LDA generates automatic summaries of topics in terms of a discrete probability distribution over words for each topic, and further infers discrete distributions of each document over topics. In detail, LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, the topic probabilities provide an explicit representation of a document. We use approximate inference techniques based on variational methods and an EM algorithm for empirical Bayesian parameter estimation. We operate as follows:

1) Choose the topic with the highest probability of each document (controversial issue) from the doc-topic matrix.
2) Add up the number of topics with the highest probability of this cluster, and select the topic with the most occurrences.
3) The top n words corresponding to this topic are obtained from the topic-words matrix as the label of the cluster.

Here, doc-topic matrix illustrates the probability distribution of the topics present in the document. Similarly, topic-words matrix illustrates the probability distribution of words generated from that topic.

After this phase, each cluster of controversial issues get its own label. Experts can quickly access cluster topics without having to read them verbatim, saving time and easing workloads. The introduction of cluster labeling plays a vital role in auxiliary data annotation.

IV. FEW-SHOT LEARNING

Even with large-scale labeled datasets, there are still many restricted in multiple aspects, as the categories of controversial issues cannot be exhausted and not all controversial issues in judgements are within the labels provided by manual annotation. In such cases, we propose to use few-shot learning to overcome this problem.

Humans have shown a strong ability to understand and recognize new concepts quickly. However, this type of generalization is not necessarily an inherent property in models, since they may fit the available classes well without learning useful structure for other classes. The process of making predictions in machine learning applications can be very computationally expensive and may become difficult with little available supervised data. Few-shot learning is a prototypical example of this setting. Few-shot learning solves the target task by learning features of a specific domain or generating inference procedures with highly discriminative properties. Given only a few examples, it makes predictions correctly without extensive retraining. As it extracts some transferable knowledge through meta-learning, few-shot learning avoid overfitting caused by applying traditional deep learning methods on sparse data tasks.

In principle, we can train a classifier to output a class label $\hat{y}$ with each test case $\hat{x}$. It works well when dealing with

similar instances, but it often fails when solving other types of problems. According to the ultimate goal of producing classifiers for a disjoint set of new classes, meta-learning was performed on the training set. In C-way G-shot problem, training episodes are formed by randomly choosing C classes from the training set with G labelled samples for each of these classes to act as the support set $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{m}$ $(m = G \times C)$ and a subset of the remainder to serve as query set $\mathcal{Q} = \left\{\left(x_j, y_j\right)\right\}_{j=1}^{n}$. The support set $\mathcal{S}$ is then fed into the model and the parameters are updated by minimizing the loss of its predictions for the examples in the query set $\mathcal{Q}$ [28].

A. RELATED MODEL

1) SIAMESE NETWORK

Koch presented a method learning siamese neural networks for one-shot image recognition [29]. This approach automatically obtaining features that enable the model to successfully generalize from a small number of examples, while limiting assumptions about the structure of the inputs. In the early 1990s, Bromley [30] first introduced siamese nets to address signature verification tasks. The siamese neural network consists of twin networks that accept different inputs but are connected by an energy function, which calculates some metrics between the highest level feature representations on each side, at the top.

Koch proposed a siamese neural network with L fully-connected layers each with N_l units as their standard model. Here, $\mathbf{h_{1,l}}$ denotes the hidden vector in layer l for the first twin while $\mathbf{h_{2,l}}$ for the second. In the first $L-1$ layers, they used exclusively rectified linear units (ReLU) [31], which means for any layer $l \in \{1, \ldots, L-1\}$:

$$\begin{aligned} h_{1,m} &= \max\left(0, \mathbf{W}_{l-1,l}^{T}\mathbf{h}_{1,(l-1)} + \mathbf{b}_l\right) \\ h_{2,m} &= \max\left(0, \mathbf{W}_{l-1,l}^{T}\mathbf{h}_{2,(l-1)} + \mathbf{b}_l\right), \end{aligned} \tag{4}$$

where $\mathbf{W}_{l-1,l}$ represents the $N_{l-1} \times N_l$ shared weight matrix connecting the N_{l-1} units in layer $l-1$ to the N_l units in layer l, and $\mathbf{b}_l$ is the shared bias vector for layer l.

After the $(L-1)$th feed-forward layers, the weighted L_1 distance between $\mathbf{h_{1,l}}$ and $\mathbf{h_{2,l}}$ were computed to compare the twin feature vectors:

$$p = \sigma\left(\sum_j \alpha_j \left|\mathbf{h}_{1,l}^{(j)} - \mathbf{h}_{2,l}^{(j)}\right|\right). \tag{5}$$

Here, α_j denotes additional parameters learned during training phase, which weighting the importance of the component-wise distance. In the last layer, a metric is induced on the learned feature space of the $(L-1)$th hidden layer and the similarity between the two feature vectors is scored.

2) MATCHING NETWORK

Motivated by the same setting of learning a class from a few labelled examples, Vinyals proposed matching nets [32], a neural network which achieve rapid learning by utilizing recent advances in attention and memory. Given a support set of k examples of input-label pairs $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{k}$, they defined the mapping $\mathcal{S} \rightarrow c_{\mathcal{S}}(\hat{x})$ to be $P(\hat{y}|\hat{x}, \mathcal{S})$. Here, $\hat{x}$ denotes a test case, $\hat{y}$ represents the appropriate label distribution for each $\hat{x}$ and P is parameterised by a neural network.

The probability over $\hat{y}$ can be obtained by:

$$P(\hat{y}|\hat{x}, \mathcal{S}) = \sum_{i=1}^{k} a\left(\hat{x}, x_i\right) y_i. \tag{6}$$

Noting that a is an attention mechanism, which uses the softmax over cosine distance with embedding functions f and g being appropriate neural networks to embed $\hat{x}$ and x_i:

$$a\left(\hat{x}, x_i\right) = e^{c(f(\hat{x}), g(x_i))} / \sum_{j=1}^{k} e^{c(f(\hat{x}), g(x_j))} \tag{7}$$

An LSTM with read-attention over the whole support set $\mathcal{S}$ can make f depend on $\hat{x}$ and $\mathcal{S}$. The encoding is the last hidden state of the LSTM. In this way, we allow the network to change its encoding of the test examples based on the training examples.

3) PROTOTYPICAL NETWORK

Prototypical Networks [16] was formulated by Snell based on the idea that classification can be performed by calculating distances to prototype representations of each class and finding the nearest class prototype for an embedded query point.

Few-shot prototypes $\mathbf{c}_k$ are computed as the mean of embedded support examples for each class through an embedding function f_ϕ with learnable parameters ϕ:

$$\mathbf{c}_k = \frac{1}{|\mathcal{S}_k|} \sum_{(\mathbf{x}_i, y_i) \in \mathcal{S}_k} f_\phi\left(\mathbf{x}_i\right) \tag{8}$$

where S_k is the set of examples labeled with class k. The distribution of classes for a query point $\mathbf{x}$ can be generated through a softmax over distances to the prototypes in the embedding space:

$$p_\phi(y = k|\mathbf{x}) = \frac{\exp\left(-d\left(f_\phi(\mathbf{x}), \mathbf{c}_k\right)\right)}{\sum_{k'} \exp\left(-d\left(f_\phi(\mathbf{x}), \mathbf{c}_{k'}\right)\right)}. \tag{9}$$

B. RELATION NETWORK

Previous studies of few-shot work used fixed pre-specified distance metrics such as Euclidean or cosine distance to perform classification [16], [32]. But it is not known whether these fixed metrics are suitable. Sung learned a good metric in a data driven way without choosing the right metric manually through using a flexible function approximator to learn similarity [28]. Relation Network can be viewed as both learning a deep embedding and learning a deep non-linear metric. It consists of two modules: an embedding module f_φ and a relation module g_ϕ.

Through f_φ, query points x_j and support samples x_i transformed to feature maps $f_\varphi\left(x_i\right)$ and $f_\varphi\left(x_j\right)$, which combined with operator $\mathcal{C}\left(f_\varphi\left(x_i\right), f_\varphi\left(x_j\right)\right)$. By feeding the combined

feature map of the sample and query into the relation module g_ϕ, it can eventually produces a scalar in range of 0 to 1. This scalar, which they called relation score, represents the similarity between x_i and x_j.

In the C way one-shot setting, they generated C relation scores $r_{i,j}$ between one query point x_j and support samples x_i,

$$r_{i,j} = g_\phi\left(\mathcal{C}\left(f_\varphi\left(x_i\right), f_\varphi\left(x_j\right)\right)\right). \quad i = 1, 2, \ldots, C \tag{10}$$

And for G-shot where $G > 1$, they summed up the embedding module outputs of all samples from each training class to form the feature map of this class. In training phase, Mean square error (MSE) loss were used to regress the relation score:

$$\varphi, \phi \leftarrow \underset{\varphi,\phi}{\operatorname{argmin}} \sum_{i=1}^{m}\sum_{j=1}^{n}\left(r_{i,j} - \mathbf{1}\left(y_i == y_j\right)\right)^2 \tag{11}$$

In our application scenario, the Chinese BERT act as the encoder. We calculated the mean of the sample vectors in the support set and concatenated them with the query points to establish the relation input. In relation module, a fully-connected layer is activated by a non-linear activation function ReLU. And the output layer is a fully connected layer activated by a sigmoid function, which ensures that relation score is produced within a reasonable range.

C. INDUCTION NETWORK

Several studies combined non-parametric methods and metric learning to provide possible solutions to few-shot learning problems [16], [28], which can rapidly assimilated new cases without suffering from catastrophic overfitting and only need to learn the representation of the samples and the metric measure. However, most class-level representations are computed by simply averaging or summing up representations of support samples. But due to the noise caused by various forms of samples in the same class, key information may be lost with the accumulation of irrelevant information. Geng proposed an Induction Network based on a class-wise level [33], which combines the dynamic routing algorithm [34] with the typical meta-learning framework. It contains three modules: encoder module, induction module and relation module.

Encoder module is a bidirectional LSTM with self-attention mechanism [35]. For a sequence of word embeddings $x = (w_1, w_2, \ldots, w_T)$, they first utilized a bidirectional LSTM to process the text:

$$\begin{aligned}\overrightarrow{h_t} &= \overrightarrow{LSTM}\left(w_t, h_{t-1}\right)\\ \overleftarrow{h_t} &= \overleftarrow{LSTM}\left(w_t, h_{t+1}\right)\end{aligned} \tag{12}$$

And then the hidden stat h_t are obtained by concatenating $\overrightarrow{h_t}$ with $\overleftarrow{h_t}$. They noted the hidden states as $H = (h_1, h_2, \ldots, h_T)$. By choosing a linear combination of the T LSTM hidden vectors in H, they encoded a variable length of text into a fixed size embedding. Taken H as the input, attention score was provided through self-attention mechanism:

$$a = \operatorname{softmax}\left(W_{a2}\tanh\left(W_{a1}H^T\right)\right) \tag{13}$$

where $W_{a1} \in R^{d_a \times 2\,u}$ and $W_{a2} \in R^{d_a}$ are weight matrixes, u is the hidden state size for each LSTM and d_a is a hyperparameter. Finally, the text representation e was obtained by the weighted sum of H:

$$e = \sum_{t=1}^{T} a_t \cdot h_t \tag{14}$$

The main idea of the induction module is to design a non-linear mapping from sample vectors ${e^s}_{ij}$ to class vector c_i.

$$\left\{e_{ij}^s \in R^{2u}\right\}_{i=1,\ldots,C,j=1,\ldots,G} \mapsto \left\{c_i \in R^{2u}\right\}_{i=1}^{C} \tag{15}$$

Here, vectors e obtained from the support set $\mathcal{S}$ are denoted as sample vectors e^s while those obtained from the query set $\mathcal{Q}$ are represented by query vectors e^q. They applied the dynamic routing algorithm in induction module, where the number of the output capsule is one. All the sample vectors share the same transformation weights $W_s \in R^{2\,u\times 2\,u}$, in order to guarantee that the model is flexible enough to accept *any*-way *any*-shot inputs. The sample prediction vector $\hat{e}_{ij}^s$ captured crucial semantic relationships between lower and higher level class features and is computed as follows:

$$\hat{e}_{ij}^s = W_s e_{ij}^s \tag{16}$$

In each iteration of dynamic routing, "routing softmax" dynamically amends the connection strength to make sure the coupling coefficients d_i sum to 1 between class i and all support samples in this class:

$$d_i = \operatorname{softmax}\left(b_i\right) \tag{17}$$

Here, b_i is the logits of coupling coefficients, which initialized by 0 in the first iteration. Each class vector $\hat{c}_i$ is calculated by:

$$\hat{c}_i = \sum_j d_{ij} \cdot \hat{e}_{ij}^s \tag{18}$$

Then they employed a non-linear "squashing" function which decreases its magnitude:

$$c_i = \frac{\left\|\hat{c}_i\right\|^2}{1 + \left\|\hat{c}_i\right\|^2}\frac{\hat{c}_i}{\left\|\hat{c}_i\right\|} \tag{19}$$

In this way, short vectors were shrunk to almost zero while long vectors shrunk slightly below one. Finally, the logits of coupling coefficients b_{ij}in every iteration is updated by a "routing by agreement" method:

$$b_{ij} = b_{ij} + \hat{e}_{ij}^s \cdot c_i \tag{20}$$

In next procedure, they measured the correlation between each pair of query and class. The correlation between c_i and e^q is called the relation score, which is the output of the Relation Module. In particular, they chose neural tensor layer [36] as an interaction function:

$$v\left(c_i, e^q\right) = f\left(c_i^T M^{[1:h]} e^q\right) \tag{21}$$

where $M^k \in R^{2u \times 2u}, k \in [1, \ldots, h]$ is one slice of the tensor parameters and f represents ReLU. The relation score r_{iq} between the i-th class and the q-th query is computed as follows:

$$r_{iq} = \text{sigmoid}\left(W_r v\left(c_i, e^q\right) + b_r\right) \tag{22}$$

Like Relation Network, they used MSE loss to train their model. There is no fine-tuning phase on the classes since the induction and comparison ability have been accumulated in the model during the training episodes.

In our research, we implemented the same encoder module as Relation Network, namely Chinese BERT. As for relation module, we applied the same settings as described in [33].

V. EXPERIMENTS

A. PRE-PROCESSING EXPERIMENTAL SETUP

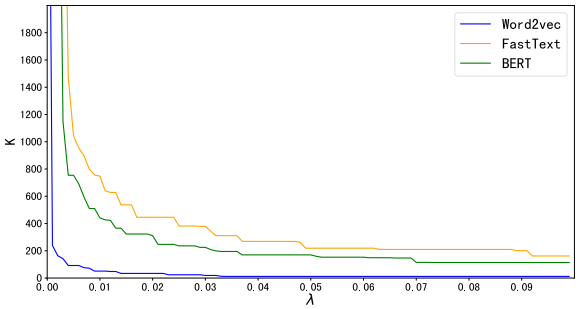
As mentioned in Section II, we conducted pre-process respectively on 1295 classes with 5521 cases in G4 of labour disputes and 241 classes with 2570 cases in G4 of product liability disputes from 2014 to 2018.

In pre-processing phase, we implemented K-means algorithm on Word2vec embeddings to set up the baseline for controversial issues merging. We improved the baseline by getting sentence embeddings directly from BERT and fastText, and adopting hierarchical clustering algorithms with Ward's distance calculation method as discussed in Section III-B. Preparation procedures of Word2vec and fastText consist of tokenization with a legal dictionary from Shaanxi People's Publishing House and the Tsinghua University Open Chinese Lexicon (THUOCL).

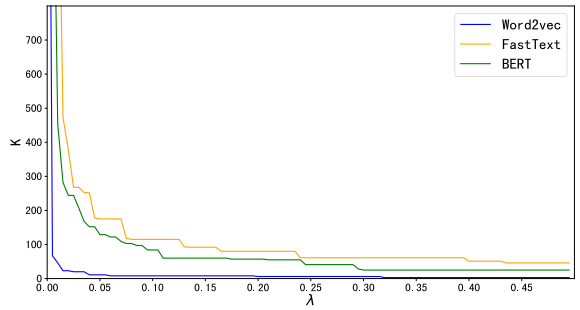
There are several parameters that need to be determined in our experiments of text clustering and cluster labeling. As illustrated in Figure3a and Figure3b, for Word2vec, fastText and BERT, we finally chose $\lambda = 0.001, 0.017, 0.031$ for labour disputes and $\lambda = 0.004, 0.075, 0.0110$ for product liability disputes with equation 3. However, sine the "long tail" contains very few controversial issues, the value of K selected by our proposed automatic method is generally lower than given manually. For symmetric Dirichlet priors in the LDA estimation, we used $\alpha = 0.1, 0.2$ and $\eta = 0.05, 0.01$, which are reasonable settings in our research.

B. FEW-SHOT LEARNING EXPERIMENTAL SETUP

To demonstrate performance of few-shot learning in controversial issues classification, we compared Relation Network and Induction Network with other classification algorithms on the same datasets. From the original Relation Network in the image field to the current Induction Networks in the NLP field, the performance of models improves with the development of few-shot learning. In the latest paper of Induction Networks, authors verified the effectiveness of Relation Network and Induction networks based on experiments on multiple datasets. So we choose these two state of the art models for text classification. We utilized SVM and TextCNN as baseline. The sentence embeddings of legal texts are all obtained by BERT. Nowadays, pretraining language



(a) Labour Disputes



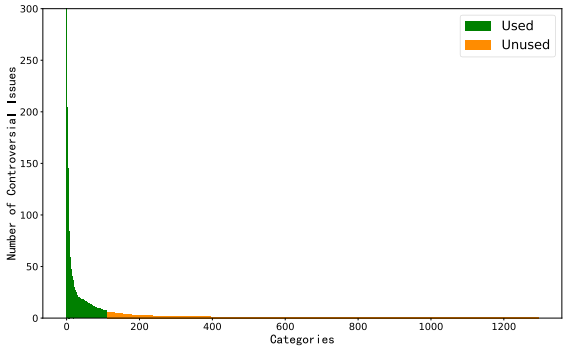
(b) Product liability Disputes

FIGURE 3. Automatic Method to Choose the optimal number of clusters K in hierarchical clustering. The "knee" in each curve is chosen as the optimal K – the point where the successive decreases in K become noticeably smaller.

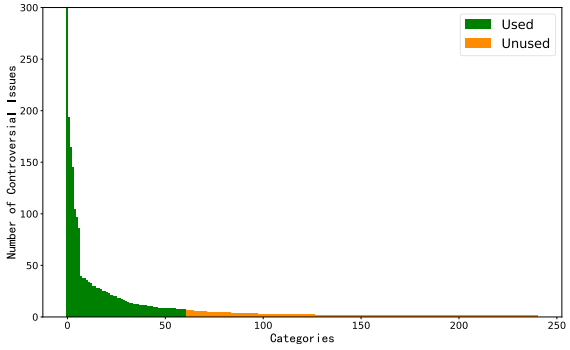
models have proven effective in NLP and can accurately capture the deep semantic information. To this end, we use the classic pretraining model in encoder module, which can get the precise semantic embedding with only a few samples. Pretraining model indeed brought an increase in the size of parameters space, but fortunately, it did not seriously reduce the predictive speed.

Analysis in Section II reflects that controversial issues follow the power-law distribution. As illustrated in Figure 4a and 4b, only classes with more than 6 samples were selected, omitting the "long tail". Therefore, 111 classes with 3795 cases of labour disputes and 61 classes with 2249 cases of product liability disputes remained for controversial issues datasets construction, as shown in the green parts of two figures. It is necessary to note that due to the extremely small amount of controversial issues in "long tail", it is difficult to display them in figures clearly. So we truncated the number to 300 in order to ensure the clarity of figures. The factual number of controversial issues in the first classes for labour disputes and product liability disputes are 859 and 558 respectively. In labour disputes, 3375 controversial issues act as training set while the remaining 420 as testing set. And in product liability disputes, 2000 controversial issues make up the training set and the remainder, 249 cases, constitute the testing set.

A combination of random hyperparameter search and artificial experience is used in tuning the hyperparameters. As for



(a) Labour Disputes. The green parts take up 8.57% of clusters and 68.74% of controversial issues.



(b) Product liability Disputes. The green parts take up 25.31% of clusters and 87.51% of controversial issues.

FIGURE 4. The usage of controversial issues. Only classes with more than 6 cases were selected, shown in green. The yellow parts of the figures represent the omitted "long tail". In order to allow the extremely small number of controversial issues in "long tail" to be clearly displayed in figures, we truncated the number to 300. The actual number of controversial issues in the first classes for labour disputes and product liability disputes are 859 and 558 respectively.

hyperparameters in Relation Network, both two causes of the action get the same settings. The relation size is 64. We built C-way G-shot models with $C = 5$ and $G = 5$, and the number of query instance in query set $\mathcal{Q}$ is 1. The train episodes is 7000. We chose Adam as the optimizer. Learning rate, warmup rate and weight decay were respectively chosen to be 2×10^{-5}, 0.06 and 0.01.

However, there are some differences in the hyperparameter settings of two causes of the action in Induction Network. For labour disputes, we chose the induction iteration number $iters = 3$. The relation size is 100. We also used $C = 5$, $G = 5$ and $\mathcal{Q}$ has 1 query instance. The train episodes is 100000. We chose the same optimizer as Relation Network. Learning rate, warmup rate and weight decay were set to be 5×10^{-5}, 0.06 and 0.01. For product liability disputes, except changing the relation size to 64, reducing train episodes to 30000 and modifying learning rate to 2×10^{-5}, other settings were the same as labour disputes.

It is also important to note that during the actual few-shot learning inference, all classes should be considered as candidate classes without prior knowledge, and support set $\mathcal{S}$ is composed by all classes from the training set with G samples for each classes. The model chooses one of the candidate classes as the classification result for each prediction. However, it may lead to expensive, slow and inference difficulty due to the large number of candidate classes. Things can be different with the joint of prior knowledge. In scenario of controversial issues, the modified cluster labels act as prior knowledge. In data annotation, manually adjustments were made to cluster labels of the original clusters. Some new clusters were added in this procedure, whose cluster labels were extracted by LDA. After these procedure, we got the modified cluster labels. A class is considered as a candidate class if its cluster labels overlap with the query. For a new query, the above problems can be solved by filtering out some portions of impossible classes based on prior knowledge, and sampling the remainder as support set.

C. METRICS

Here we discuss the associated evaluation metrics. As for the evaluation of clustering, we use a set of classes in an evaluation benchmark. Then we can compute the criterion that evaluates how well the clustering matches the standard classes. Adjusted mutual information (AMI) [37] is the metric most often applied, together usually, with V-measure [38]. We utilize accuracy [39], macro F_1 [40] and weighted F_1 to evaluate the results of few-shot learning experiments.

D. RESULTS

In this section we show the results of our experiments. We are interested in answering two questions. Firstly, we want to see if our improved pre-processing methods achieves better results than the baseline methods that work under the same conditions. Secondly, we want to know if few-shot learning can perform well in legal field.

In Table 1 results of our experiments are listed. For convenience, the results of the baseline methods, as mentioned above, are displayed in the top two rows. The rows marked "manual" take into account expert opinion when choosing K, while the rows marked "heuristic" or "automatic" respectively apply the heuristic method or our proposed method in the selection of K. In the middle half of Table 1, we apply K-means with different embeddings. And in the lower half we adopt HAC. As can be seen from the table, fastText and BERT achieve improvements over K-means clustering. Considering that BERT based on K-means has already obtained significant improvements, and is therefore a higher baseline, the performance improvements of HAC are encouraging. However, as controversial issues obey the power-law distribution and include too many unbalanced classes, text clustering did not work well. There are still many controversial issues fall into the wrong places and need to be corrected manually.

The next task is to label each group of homogeneous controversial issues. Figure 5 shows an example of the results of cluster labeling based on LDA. The pivotal content of the cluster were captured and revealed in the first row. Instead of

TABLE 1. Clustering performance comparison of several methods.

Methods	Labour Disputes		Product Liability Disputes	
	AMI	$V-Measure$	AMI	$V-Measure$
Word2vec-K-means^{m}	0.1552	0.6711	0.2094	0.5235
Word2vec-K-means^{h}	0.1835	0.5677	0.2474	0.4352
fastText-K-means^{m}	0.1630	0.6671	0.2194	0.5197
fastText-K-means^{h}	0.1958	0.5676	0.2578	0.4295
BERT-K-means^{m}	0.1675	0.6740	0.2270	0.5284
BERT-K-means^{h}	0.2090	0.5787	0.2739	0.4654
Word2vec-HAC^{m}	0.3751	0.7939	0.4289	0.6895
Word2vec-HAC^{a}	0.4722	0.7271	0.4868	0.6066
fastText-HAC^{m}	0.4038	0.8034	0.4312	0.6985
fastText-HAC^{a}	0.4960	0.7294	0.4859	0.6041
BERT-HAC^{m}	0.4231	**0.8120**	0.4434	**0.7027**
BERT-HAC^{a}	**0.5250**	0.7777	**0.4901**	0.6683

^{m}manual: expert opinion of K; ^{h}heuristic: heuristic method for choosing K in flat clustering;
^{a}automatic method for choosing K in hierarchical clustering.

TABLE 2. Classification performance comparison of several methods.

Methods	Labour Disputes			Product Liability Disputes		
	$Accuracy$	$MacroF_1$	$WeightedF_1$	$Accuracy$	$MacroF_1$	$WeightedF_1$
SVM	86.43	76.24	84.28	89.16	78.4	86.45
TextCNN	86.90	69.44	84.18	88.35	75.01	85.72
Relation Network	95.24	**93.13**	95.16	**94.78**	**88.44**	**93.93**
Induction Network	**96.19**	92.63	**95.78**	93.17	88.15	92.64

Example Cluster - Product Liability Disputes

Cluster label: 是否，产品，质量，缺陷，造成
1、案涉车辆自燃是否为车辆缺陷造成的?
2、涉案车辆发生交通事故是否属于因产品质量缺陷而导致?
3、电暖宝是否因产品缺陷发生爆炸?
4、安全气囊没有张开是否属于产品质量问题造成?
5、车辆是否存在导致事故发生的质量缺陷?
6、火灾原因是否属涉案车辆产品缺陷引起的自燃?

Cluster label: whether, product, quality, defect, cause by
1、Whether the vehicle spontaneous combustion was caused by vehicle defect?
2、Whether the traffic accident of the vehicle involved is caused by defect in product quality?
3、Whether electric heater exploded due to product defect?
4、Whether the safety airbag was not deployed is caused by product quality problems?
5、Is there any quality defect in the vehicle that caused the accident?
6、Whether the cause of the fire was spontaneous combustion caused by the vehicle product defect?

FIGURE 5. Cluster labeling based on LDA.

reading text verbatim, experts can identify the main information of controversial issues in each cluster from cluster labels.

In addition to studying the findings as discussed above, it is necessary to see how the few-shot classification performs. In Table 2 we show the results, the top two rows recorded the results of SVM and TextCNN, while the last two rows showed the performances of few-shot learning. As expected, few-shot learning achieved high scores in both Relation Network and Induction Network. This indicates that few-shot learning is a successful strategy for classifying controversial issues.

VI. CONCLUSION AND FUTURE WORKS

With the gradual formation of the new procedural system, Chinese courts organize debates surrounding controversial issues. Judges divide controversial issues into the factual controversial issues and legal controversial issues in order to ascertain the facts and then carry out legal reasoning. Both of them have been thought of as essential elements in enhancing court efficiency. Further, controversial issues play key roles in trial scene and judges' decision making restoration. But due to the limited number of cases that individual judge has access to, it is difficult for them to draw on experiences of other judges in summarizing controversial issues and conducting trials. It is urgent to classify cases with different types of controversial issues. Therefore, homogeneous controversial issues classification becomes the basis for cases classification. As controversial issues follow the power-law distribution, not all of them are within the labels provided by manual annotation and their categories cannot be exhausted. In order to generalizing those unseen categories without necessitating extensive retraining, we introduce a controversial issues classification algorithm based on few-shot learning.

In preprocessing module, different from baseline model, we utilize state-of-the-art Chinese BERT pre-training model with hierarchical algorithms and evaluated the method using adjudicative documents of labour disputes and product liability disputes. Experiments have demonstrated that our improved method is superior to baseline in terms of capturing the deep semantic information of the text and enhancing the quality of grouping. After clustering phase, homogeneous controversial issues are merged into a group. The LDA topic model is adopted on each cluster to facilitate experts access cluster topics without having to read verbatim and find miscategoried controversial issues faster in data annotation. In classification module, we use two few-shot learning algorithms, Relation Network and Induction Network. Given only a handful of instances, experiment results demonstrate that both of them outperform existing classification methods significantly. The introduced method provides trial assistance for judges, which promotes the dissemination of experience and improves the fairness of adjudication.

Nevertheless, there is still room to improve our model in the future. For instance, legal corpus can be introduced in training the Encoder Chinese-BERT of few-shot learning, which will be more suitable for this scenario. Unlike the two models in the article that do not consider query information in the Induction Module, an attention induction module based on the current query can be constructed to avoid wasting information, which may also improve the effectiveness of the model. Also, as legal data accumulates, the impact of the ''long tail'' will decrease. We believe this research can lead to many fruitful studies to provide trial assistance in the Chinese legal spheres.

REFERENCES

[1] C. Gui-Ming, ''Issues concerning several relations in the design of pretrial preliminary procedure,'' *Political Sci. Law Tribure*, vol. 4, no. 11, pp. 9–15, 2004.

[2] T. Mikolov, K. Chen, G. Corrado, and J. Dean, ''Efficient estimation of word representations in vector space,'' 2013, *arXiv:1301.3781*. [Online]. Available: http://arxiv.org/abs/1301.3781

[3] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, ''Enriching word vectors with subword information,'' *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 135–146, Dec. 2017.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ''BERT: Pre-training of deep bidirectional transformers for language understanding,'' 2018, *arXiv:1810.04805*. [Online]. Available: http://arxiv.org/abs/1810.04805

[5] A. Clauset, C. R. Shalizi, and M. E. J. Newman, ''Power-law distributions in empirical data,'' *SIAM Rev.*, vol. 51, no. 4, pp. 661–703, Nov. 2009.

[6] N. Eikmeier and D. F. Gleich, ''Revisiting power-law distributions in spectra of real world networks,'' in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2017, pp. 817–826.

[7] C. Francalanci, A. Hussain, and F. Merlo, ''Representing social influencers and influence using power-law graphs,'' *Appl. Math. Inf. Sci.*, vol. 9, no. 5, p. 2453, 2015.

[8] W. A. Awad, ''Machine learning methods for spam e-mail classification,'' *Int. J. Comput. Sci. Inf. Technol.*, vol. 3, no. 1, pp. 173–184, Feb. 2011.

[9] Y. Wang, M. Huang, X. Zhu, and L. Zhao, ''Attention-based LSTM for aspect-level sentiment classification,'' in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 606–615.

[10] C. Xian-Yi, Z. Ling-ling, Z. Qian, and W. Jin, ''The framework of network public opinion monitoring and analyzing system based on semantic content identification,'' *J. Converg. Inf. Technol.*, vol. 5, no. 10, pp. 48–55, Dec. 2010.

[11] S. Lloyd, ''Least squares quantization in PCM,'' *IEEE Trans. Inf. Theory*, vol. IT-28, no. 2, pp. 129–137, Mar. 1982.

[12] X. Tian, Y. Fang, Y. Weng, Y. Luo, H. Cheng, and Z. Wang, ''K-means clustering for controversial issues merging in Chinese legal texts,'' in *Proc. JURIX*, 2018, pp. 215–219.

[13] A. K. Jain, ''Data clustering: 50 years beyond K-means,'' *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, Jun. 2010.

[14] D. M. Blei, A. Y. Ng, and M. I. Jordan, ''Latent Dirichlet allocation,'' *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[15] M. G. Miller, N. E. Matsakis, and P. A. Viola, ''Learning from one example through shared densities on transforms,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2000, pp. 464–471.

[16] J. Snell, K. Swersky, and R. Zemel, ''Prototypical networks for few-shot learning,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4077–4087.

[17] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, ''Deep contextualized word representations,'' 2018, *arXiv:1802.05365*. [Online]. Available: http://arxiv.org/abs/1802.05365

[18] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, ''Improving language understanding with unsupervised learning,'' OpenAI, San Francisco, CA, USA, Tech. Rep., Jun. 2018.

[19] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2001.

[20] E. M. Rasmussen, ''Clustering algorithms,'' *Inf. Retr., Data Struct. Algorithms*, vol. 419, p. 442, Jun. 1992.

[21] N. Jardine and C. J. van Rijsbergen, ''The use of hierarchic clustering in information retrieval,'' *Inf. Storage Retr.*, vol. 7, no. 5, pp. 217–240, Dec. 1971.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, ''Maximum likelihood from incomplete data via the em algorithm,'' *J. Roy. Stat. Soc., B Methodol.*, vol. 39, no. 1, pp. 1–22, 1977.

[23] H. Schütze, C. D. Manning, and P. Raghavan, *Introduction to Information Retrieval*, vol. 39. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[24] B. King, ''Step-wise clustering procedures,'' *J. Amer. Stat. Assoc.*, vol. 62, no. 317, pp. 86–101, Mar. 1967.

[25] P. H. A. Sneath and R. R. Sokal, *Numerical Taxonomy. The Principles and Practice of Numerical Classification* 1973.

[26] G. N. Lance and W. T. Williams, ''A general theory of classificatory sorting strategies: 1. Hierarchical systems,'' *Comput. J.*, vol. 9, no. 4, pp. 373–380, Feb. 1967.

[27] A. El-Hamdouchi and P. Willett, ''Hierarchic document classification using Ward's clustering method,'' in *Proc. 9th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 1986, pp. 149–156.

[28] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, ''Learning to compare: Relation network for few-shot learning,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.

[29] G. Koch, R. Zemel, and R. Salakhutdinov, ''Siamese neural networks for one-shot image recognition,'' in *Proc. ICML Deep Learn. Workshop*, vol. 2, pp. 1–27, 2015.

[30] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, ''Signature verification using a 'Siamese' time delay neural network,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 737–744.

[31] X. Glorot, A. Bordes, and Y. Bengio, ''Deep sparse rectifier neural networks,'' in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.

[32] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, ''Matching networks for one shot learning,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3630–3638.

[33] R. Geng, B. Li, Y. Li, X. Zhu, P. Jian, and J. Sun, ''Induction networks for few-shot text classification,'' 2019, *arXiv:1902.10482*. [Online]. Available: http://arxiv.org/abs/1902.10482

[34] S. Sabour, N. Frosst, and G. E. Hinton, ''Dynamic routing between capsules,'' in *Proc. Adv. neural Inf. Process. Syst.*, 2017, pp. 3856–3866.

[35] W. Lin, C. Zhang, K. Lu, B. Sheng, J. Wu, B. Ni, X. Liu, and H. Xiong, ''Action recognition with coarse-to-fine deep feature integration and asynchronous fusion,'' in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 7130–7137.

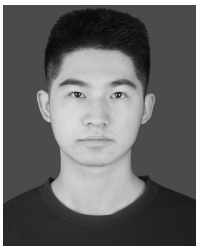
[36] R. Socher, D. Chen, C. D. Manning, and A. Ng, ''Reasoning with neural tensor networks for knowledge base completion,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 926–934.

[37] N. X. Vinh, J. Epps, and J. Bailey, ''Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance,'' *J. Mach. Learn. Res.*, vol. 11, pp. 2837–2854, Jan. 2010.

[38] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn. (EMNLP-CoNLL)*, 2007, pp. 410–420.
[39] P. Annesi, D. Croce, and R. Basili, "Semantic compositionality in tree kernels," in *Proc. 23rd ACM Int. Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2014, pp. 1029–1038.
[40] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity," in *Proc. AAAI*, vol. 6, 2006, pp. 775–780.

YIN FANG received the B.S. degree from Sichuan University, Chengdu, China, in 2017. She is currently a Graduate Student with the Department of Mathematics, Sichuan University. Her current research interests include Bayesian non-parametrics and natural language processing.

XIN TIAN received the B.S. degree from Zhengzhou University, Zhengzhou, China, in 2017. He is currently a Graduate Student with the Department of Mathematics, Sichuan University, Chengdu, China. His research interests lie within the intersection of natural language processing and machine learning.



HAO WU received the B.S. degree from Sichuan University, Chengdu, China, in 2018. He is currently a Graduate Student with the Department of Mathematics, Sichuan University. His current research interests include knowledge graphs and statistic machine learning.

SONGYUAN GU received the LL.B. degree from Sichuan University, Chengdu, China, in 2019. She is currently a Graduate Student with the Law School, Sichuan University. Her current research interests include civil law and big data analysis of law.

ZHU WANG received the B.B.A. degree from the Department of Commodity Science, Renmin University of China, in 2003, and the LL.M. and LL.D. degrees from the Law School, Renmin University of China, in 2006 and 2009, respectively. He is currently a Professor of the Law School, PI of LAIW (AI in LAW) Advanced Deployed Discipline, the Director of the Institute of Rule of Law of Market Economy, and a Researcher of the Institute for Statistic, Sichuan University. His research focuses on tort, insurance law, and constitution and big data analysis of law.

FENG WANG received the master's degree from the Southwestern University of Finance and Economics, Chengdu, China, in 2017. He is currently an Algorithm Researcher with Union Big Data Technology. His current work focuses on the application of NLP in the field of law.

JUNLIANG LI received the B.S. degree major in computer science from the University of Electricity Science and Technology of China, Chengdu, China, and the master's degree major in M.B.A. from Sichuan University, Chengdu, China. He has been worked in legal technology industry for a long time.

YANG WENG received the B.S. and Ph.D. degrees from the Department of Mathematics, Sichuan University, Chengdu, China, in 2001 and 2006, respectively. Since 2006, he has been with the College of Mathematics, Sichuan University, where he is currently an Associate Professor. He was a Postdoctoral Fellow with the Nanyang Technological University, Singapore, from August 2008 to July 2010. His current research interests include statistic machine learning and nonparametric Bayesian inference.