# Study on Multi-Label Classification of Medical Dispute Documents

**Baili Zhang[1, 2, 3, *], Shan Zhou[1], Le Yang[1], Jianhua Lv[1, 2] and Mingjun Zhong[4]**

**Abstract:** The Internet of Medical Things (IoMT) will come to be of great importance in the mediation of medical disputes, as it is emerging as the core of intelligent medical treatment. First, IoMT can track the entire medical treatment process in order to provide detailed trace data in medical dispute resolution. Second, IoMT can infiltrate the ongoing treatment and provide timely intelligent decision support to medical staff. This information includes recommendation of similar historical cases, guidance for medical treatment, alerting of hired dispute profiteers etc. The multi-label classification of medical dispute documents (MDDs) plays an important role as a front-end process for intelligent decision support, especially in the recommendation of similar historical cases. However, MDDs usually appear as long texts containing a large amount of redundant information, and there is a serious distribution imbalance in the dataset, which directly leads to weaker classification performance. Accordingly, in this paper, a multi-label classification method based on key sentence extraction is proposed for MDDs. The method is divided into two parts. First, the attention-based hierarchical bi-directional long short-term memory (BiLSTM) model is used to extract key sentences from documents; second, random comprehensive sampling Bagging (RCS-Bagging), which is an ensemble multi-label classification model, is employed to classify MDDs based on key sentence sets. The use of this approach greatly improves the classification performance. Experiments show that the performance of the two models proposed in this paper is remarkably better than that of the baseline methods.

**Keywords:** Internet of Medical Things (IoMT), medical disputes, medical dispute document (MDD), multi-label classification (MLC), key sentence extraction, class imbalance.

## 1 Introduction

Nowadays, escalating medical disputes have impeded regular medical treatment and even caused social turmoil. Researchers have thus endeavored to improve the medical management system and explore effective methods of medical dispute resolution with the

[1] School of Computer Science and Engineering, Southeast University, Nanjing, 21189, China.

[2] Key Laboratory of Computer Network and Information Integration of Ministry of Education, Southeast University, Nanjing, 211189, China.

[3] Research Center for Judicial Big Data, Supreme Count of China, Nanjing, 211189, China.

[4] University of Lincoln, Lincoln, LN6 7TS, UK.

[*] Corresponding Author: Baili Zhang. Email: zhangbl@seu.edu.cn.

Received: 06 April 2020; Accepted: 17 July 2020.

help of advanced technologies, including artificial intelligence (AI), the Internet of Things (IoT), the Internet of Medical Things (IoMT) and blockchain [Al-Fuqaha, Guizani, Mohammadi et al. (2015); Gharaibeh, Salahuddin, Hussini et al. (2017); Wang, Kong, Guan et al. (2019); Wang, Kong, Li et al. (2019)]. Of these, IoMT in particular is becoming increasingly mature and prominent [He, Ye, Chan et al. (2018); He, Xie, Xu et al. (2019); Zhang and Zhou (2014)]. Through the use of sensors, radio frequency identification (RFID), data processing, video detection and identification technology, IoMT is able to realize the intelligent identification, positioning, tracking and monitoring of people, events and behaviors in a medical treatment process; in so doing, it gradually establishes a real-time, accurate and efficient intelligent medical system. Accordingly, IoMT is able to provide a satisfactory platform for the effective resolution of medical disputes. Firstly, it is able to provide true and detailed trace data for use in medical dispute resolution, which can help mediators reproduce the entire medical process more accurately; secondly, it can infiltrate the ongoing treatment process and provide timely intelligent decision support, such as the recommendation of similar historical cases, guidance for medical treatment, alerting of hired dispute profiteers etc. As the most representative tool for the mediation of medical disputes, recommendation of similar historical cases can present similar cases to the medical staff for reference at the appropriate time. If medical dispute documents are classified prior to recommendation, it is possible to quickly filter out those historical cases that are inconsistent with the categories of user input cases, thus improving the accuracy and efficiency of the document recommendation. In short, the classification of medical dispute documents can play an important role as a front-end process for intelligent decision support, especially in the recommendation of similar historical cases. "Medical negligence behavior" causing harm to patients, is one of the most important categories in MDD. Since a document may be associated with multiple "medical negligence behavior"-related labels, the classification of MDDs studied in this paper is essentially a multi-label classification (MLC) problem.

At present, problem transformation methods (including binary relevance (BR), label powset (LP), random k-labelsets (RAKEL) and classifier chain (CC)), algorithm adaptation methods (including Rank-SVM and ML-KNN), and neural network MLC models are widely applied in the MLC context [Zhang and Zhou (2014); Nam, Kim, Mencía et al. (2013); Kurata, Xiang and Zhou (2016)]. Experiments demonstrate that these methods are generally able to achieve good results for short and medium-length texts of less than 500 words on average, but do not work well for long MDDs. In these long documents, a lot of redundant information needs to be incorporated for the sake of professional expression, while the content related to the classification themes is scattered throughout the text. These interferences make it difficult for the model to capture the features of the information related to the classification.

In addition, MDD datasets suffer from a noticeable distribution imbalance, which impairs the effectiveness of the classification model. Currently available solutions to the dataset imbalance problem include resampling and algorithm adaptation. As resampling is independent of the classifier, it has been widely employed, with many related algorithms having been studied (including random under-sampling (LP-RUS), random over-sampling (LP-ROS), random under-sampling (ML-RUS) and random over-sampling (ML-ROS)) [Charte, Rivera, María del Jesus et al. (2015)]. However, none of these approaches can either distinguish between the effects of the samples with different label sets on solving the dataset imbalance or reduce the impact of imbalance label co-occurrence on resampling.

To deal with the above-mentioned problems, this paper proposes an MLC method based on key sentence extraction for MDDs. The proposed approach is divided into two steps. First, the attention-based hierarchical BiLSTM model is used to extract key sentences from MDDs in order to filter out irrelevant information. Second, RCS-Bagging, a new MLC model, is adopted to classify MDDs on the basis of key sentence sets. The proposed model combines resampling and Bagging in order to reduce the impact of the unbalanced distribution of labels, and further improves the classification performance.

## 2 Feature analysis of MDD datasets

Unlike general texts, MDDs are characterized by a complete structure, rigorous logic and remarkable length. The repetition of certain facts and explanations across different paragraphs leads to a large amount of redundant information being contained in MDDs. Statistics reveal that around 6.4% of MDD texts exceed 10,000 words in length, while only 2% are under 1,000 words; moreover, 58% are between 1,000-5,000 words in length, and 33.6% are between 5,000-10,000 words. For instance, content related to "medical negligence behavior" will likely be scattered across four different paragraphs pertaining to the "plaintiff's claim", "defendant's argument", "facts" and "court's opinions". As a result, a large amount of redundant and irrelevant information is present in these texts, which undermines the classifier performance.

As linguistic units capable of expressing relatively complete meaning, sentences can clearly describe medical negligence behavior. In our work, the sentences related to this theme are first extracted from a text so that the content irrelevant to the classification can be eliminated. Subsequently, MLC is employed to classify MDDs on the basis of the key sentences. By extracting the key sentences, the classifier can accurately grasp the main concepts in the text, meaning that its performance can be considerably improved.

The distribution of "medical negligent behavior types" in MDDs is illustrated in Fig. 1. Some labels appear frequently, while others appear infrequently. However, the phenomenon of class imbalance causes the results of the MLC algorithms to be unsatisfactory. Therefore, the resampling and ensemble techniques are adopted in this paper to reduce the impact of class imbalance and thereby improve the classification performance.

## 3 The attention-based Hierarchical BiLSTM model

The proposed model consists of a sentence encoder and a sentence extractor. The sentence encoder maps the sentence to a fixed-length vector in order to obtain the sentence's semantic representation vector. The sentence extractor is used to calculate the probability of a certain sentence becoming a key sentence, then extract the key sentences. The overall architecture of the model is illustrated in Fig. 2.

### *3.1 Encoder*

Yang et al. [Yang, Yang, Dyer et al. (2016)] use BiLSTM [Hochreiter and Schmidhuber (1997)] to obtain word representations; a similar approach is used in this paper. Assuming that a document has $L$ sentences, each sentence $S_i$ contains $T_i$ words. $w_{it}$ represents the $t$-th word in the $i$-th sentence and can be embedded into vectors through the word embedding matrix $W_e$; i.e., $x_{it} = W_e w_{it}, t \in [1, T_i]$, BiLSTM is used to obtain word representations by summarizing information about the words from both directions for words: $\vec{h} = l\vec{stm}(x_{it}), t \in [1, T_i]$, $\overleftarrow{h} = lst\overleftarrow{m}(x_{it}), t \in [1, T_j]$. A representation is obtained for a given word $w_{it}$ by concatenating the forward hidden state $\vec{h}_{it}$ and backward state $\overleftarrow{h}_{it}$, i.e., $h_{it} = [\vec{h}_{it}; \overleftarrow{h}_{it}]$, which summarizes the information of the entire sentence centered around $w_{it}$.

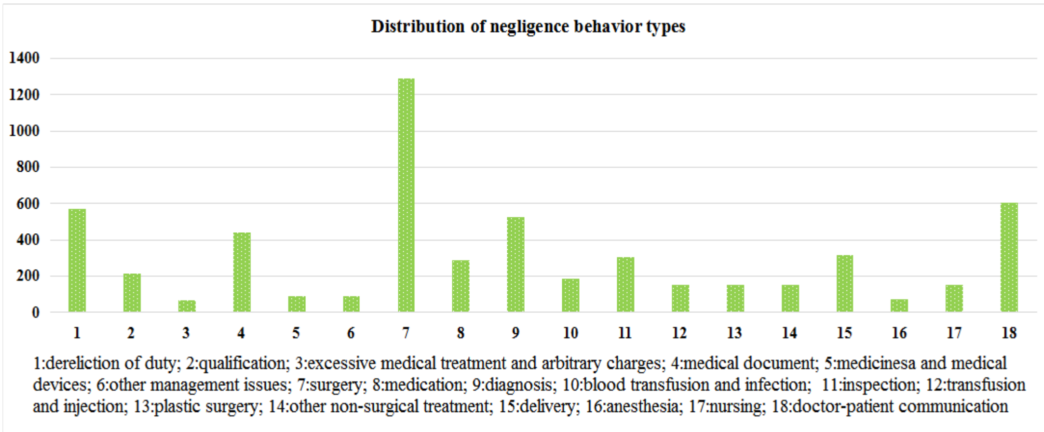


**Figure 1:** Distribution of negligent behavior types

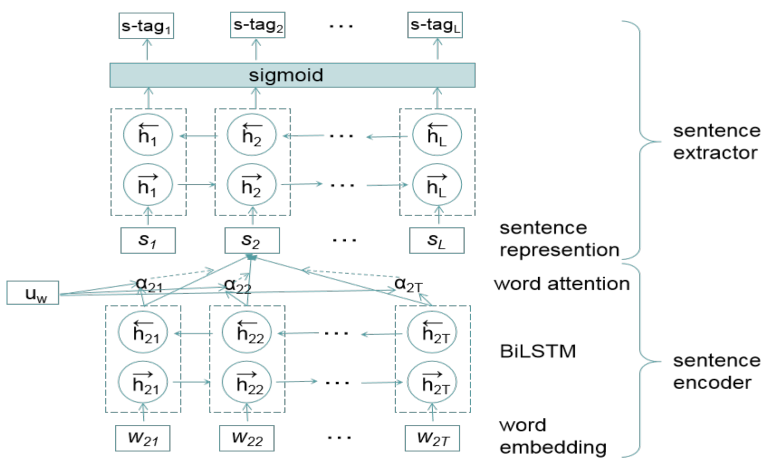


**Figure 2:** Attention-based hierarchical BiLSTM model

Next, the attention mechanism [Bahdanau, Cho and Bengio (2014)] is introduced. The purpose of this mechanism is to extract key words and aggregate the representation of those informative words to form a sentence vector. More specifically, the attention mechanism is used to generate a sentence vector $s_i$, which is computed as a weighted sum of word representations and obtained via Eq. (3):

$$u_{it} = \tanh(W_e h_{it} + b_w) \tag{1}$$

$$\alpha_{it} = \exp(u_{it}^T u_w) / \sum_t \exp(u_{it}^T u_w) \quad (2)$$

$$s_i = \sum_t \alpha_{it} h_{it} \quad (3)$$

Here, $u_{it}$ is the word context vector, which is randomly initialized and updated by Eq. (1). Moreover, $\alpha_{it}$ is a weighted vector and obtained by Eq. (2).

### *3.2 Extractor*

The extractor also contains a BiLSTM layer, which is used to annotate the sentence sequence. Similarly, given the sentence vectors $S_j$, BiLSTM is used to encode the sentences, as follows: $\vec{h} = \overrightarrow{lstm}(s_i), i \in [0, L]$, $\overleftarrow{h} = \overleftarrow{lstm}(s_i), i \in [0, L]$ Subsequently, the representation of sentence $i$ is obtained by concatenating $\vec{h}_i$ and $\overleftarrow{h}_i$, i.e., $h_i = [\vec{h}_i ; \overleftarrow{h}_i], i \in [0, L]$.

With reference to the representation of sentence $h_i$, the probability of sentence $i$ being a key sentence is computed as follows: $p(s\_tag_i = 1 | D) = \sigma(Wh_i + b)$, where σ is a sigmoid function.

## 4 Multi-label classification model

Class imbalance in a dataset makes it difficult to obtain a single classifier that achieves good performance in the MLC context. Bagging [Breiman (1996)], a well-known representative of parallel ensemble learning methods, not only improves the classifier performance but also deals well with class imbalance in binary classification contexts. However, the bootstrap sampling method adopted by Bagging is unable to reduce the multi-label class imbalance ratio. Therefore, random comprehensive sampling (RCS), an improved sampling algorithm, is presented in this section. RCS is combined with Bagging to create a new algorithm, named RCS-Bagging.

### *4.1 Random Comprehensive Sampling Algorithm (RCS)*

The imbalance ratio per label (IRLbl) was proposed in the literature to measure the imbalance ratio [Wang, Kong, Li et al. (2019)]. Given a set of labels $Y = \{y_1, y_2, ..., y_q\}$ and a training set $S = \{(x_1, Y_1), ..., (x_m, Y_m)\}$, where $x_i$ is a single instance and $Y_i$ (associated with $x_i$) is a subset of *Y*, IRLbl(*y*) is computed as the imbalance ratio of the label *y*, as computed in Eq. (4):

$$IRLbl(y) = \arg\max_{y=y_1}^{y_q} \left(\sum_{i=1}^{m} h(y', \ Y_i)\right) / \sum_{i=1}^{m} h(y, Y_i) \quad (4)$$

Moreover, $h(y, Y_i)$ is computed as in Eq. (5).

$$h(y, Y_i) = \begin{cases} 1, y \in Y_i \\ 0, y \notin Y_i \end{cases} \quad (5)$$

Clearly, the larger the IRLbl, the higher the imbalance ratio of the label $y$. Based on IRLbl, the mean imbalance ratio of the label set (MeanIRls) can be defined as the average of the imbalance ratio of the labels in label set, which is computed by Eq. (6):

$$MeanIRls(labelset) = \frac{1}{p}\sum_{y=y_1}^{y_p} IRLbl(y) \tag{6}$$

Here, $p$ is the size of the label set $Y_i$. The larger the MeanIRls, the higher the overall imbalance rate of the label included in the label set $Y_i$. In this way, the contributions made by samples with different label sets can be distinguished. An improved sampling algorithm (RCS) is therefore proposed. By calculating the MeanIRls of each label set, two sets of samples (with higher and lower MeanIRls) are obtained. Finally, the samples with higher MeanIRls are randomly over-sampled, while the samples with lower MeanIRls are randomly under-sampled, resulting in the imbalance ratio of the datasets being lower than the original dataset after resampling. The pseudocode of the algorithm is as follows:

Input: Unbalanced Dataset D; OverSampling Set Proportion M, UnderSampling Set Proportion N (Determine Sampling Candidate Set Size); OverSampling Rate P, UnderSampling Rate Q (Determine Sampling Number)

Output: Approximate Balanced Data Set D'

```
    samplesToClone=|D|*P
    samplesToDelete=|D|*Q
    OSNum=|D|*M
    USNum=|D|*N
    // Get label space, labelsets and Bags of samples corresponding to the same labelset
    Labels, Labelsets, labelsetBag<-labelsInDataset(D)
    for each label in Labels do
       IRLbllabel<-calculateIRperLabel(D, label)
    for each labelset in Labelsets do
       MeanIRSlabelset<-calculateMeanIRperLabelset(labelset)sortedLabelsetBag=sorted(labelsetBag,
MeanIRls)
    //The OSNum samples with the highest MeanIRls corresponding to labelset are selected as candidate
    the USNum samples with the lowest MeanIRls are selected as candidate sets for undersampling.
       OSBag, USBag=getSamplingBag(sortedLabelsetBag, OSNum, USNum)
       while samplesToClone > 0:
       x<-random(1, |OSBag|)
       cloneSample(x, D)
       samplesToClone--
       while samplesToDelete > 0:
       x<-random(1, |USBag|)
       deleteSample(x, D)
       samplesToDelete--
```

### *4.2 RCS-Bagging*

Our work combines RCS with Bagging in order to produce an improved algorithm for MLC, named RCS-Bagging, based on key sentence sets. More specifically, the basic procedure of Bagging is adopted, while the bootstrap sampling is replaced with RCS in order to reduce the imbalance ratio in the base classifier's training set. In more detail, the process involved can be divided into three steps:

(1) Using RCS to resample the original training set and generate *n* sample sets;

(2) Training a multi-label classifier for each sampling set;

(3) Combining the *n* base classifiers.

In RCS-Bagging, the "one-vote decision" strategy is adopted to combine the classifiers; that is, a label is assigned if one or more base classifiers vote for it. Evidently, the classifier combining the "one-vote decision" strategy will result in a lower precision and a higher recall. Recalling that our aim is case recommendation, high recall can prevent the corresponding documents from being incorrectly filtered out, while lower precision can be compensated for in the text similarity calculation process. Therefore, a strategy with higher recall rate is required in our work.

## 5 Experimentation and analysis

In order to verify the effectiveness of the attention-based hierarchical BiLSTM model and the RCS-Bagging model, we conduct two sets of comparison experiments for key sentence extraction and MLC.

### *5.1 Key sentence extraction*

**Dataset:** A total of 2,000 documents were randomly selected. Of these, the longer documents were cut into multiple shorter texts of different themes (ranging from "plaintiff's claim", "defendant's claim" and "facts" to "court's opinion") to facilitate the model training. Subsequently, the key sentences (i.e. those related to negligent behavior) were labeled manually in the clipped documents to create the dataset of key sentence extraction.

**Baseline:** Key sentence extraction and text summarization are closely related to each other, having some things in common but differing in other aspects. Generally speaking, text summarization aims to express the main concept in a text using fewer words, meaning that it needs to avoid redundancy in the extracted sentences [Cao, Wei, Li et al. (2015); Nallapati, Zhai and Zhou (2016)]. However, in order to ensure that the extracted key sentence sets covers all the negligent behaviors described in the documents, the elimination of redundancy between sentences was not taken into account in our work. In our experiments, two text summarization methods that do not involve redundancy elimination were compared with the attention-based hierarchical BiLSTM model. The first is a traditional method proposed by Kupiec et al. [Kupiec and Chen (1995)], while the second is a neural summarization model proposed by Cheng et al. [Cheng and Lapata (2016)].

**Hyperparameters and training:** In our experiments, the word vectors were initialized using 50-dimensional pre-trained embedding, and were trained on all medical dispute documents using the word2vec model. The batch-size was set to 128. At the same time, the real length of the documents and sentences were stored in an additional mask variable.

The hidden size was set to 50 and the dropout rate to 0.5. The Adam optimizer (lr=0.001) was used to update the weight.

**Evaluation:** In our work, key sentence extraction was regarded as a form of sentence classification. Therefore, the model performance was evaluated in terms of accuracy, recall and F1 score, which are commonly used metrics in classification problems.

**Results and analysis:** The results of our model and those of the comparison methods are listed in Tab. 1. It is clear from the table that our attention-based hierarchical BiLSTM model significantly outperforms the other baselines on the MDD dataset. As the neural network model is able to make full use of semantic and sequence information, its performance is superior to the traditional summarization methods [Kupiec and Chen (1995); Cheng and Lapata (2016)]. The attention-based hierarchical BiLSTM model also avoids a problem that commonly occurs with Encoder-Decoder-based summarization models [Cheng and Lapata (2016); Narayan, Cohen and Lapata (2018)]: that is, because the sequence information is compressed into a fixed-length context vector in the Encoder-Decoder model, the previous information can be easily "forgotten" by the encoder as the sentence length increases, so that the context vector fails to represent the meaning of the entire input sequence.

**Table 1:** Comparison of key sentence extraction models

| | Precision | Recall | F1 |
|---|---|---|---|
| Kupiec et al. | 77.25% | 66.39% | 71.41% |
| Cheng et al. | 88.67% | 89.34% | 89.00% |
| Ours | 94.79% | 94.73% | 94.76% |

### *5.2 Multi-label classification*

**Dataset:** Following the analysis of a large number of documents, the common negligent behaviors found in MDDs were summarized into 18 categories. The documents containing key sentences were labeled with different negligent behavior types to form the MLC dataset. The specific category information is presented in Fig. 1.

**Baseline:** Preliminary experiments on the MDD dataset demonstrate that with linear kernel SVM being underlying classifier, BR, RAKEL and CC are superior to other MLC algorithms for MDDs dataset. Accordingly, these three MLC algorithms were selected as the base classifiers of Bagging. Moreover, in order to verify the effectiveness of RCS-Bagging, RCS was compared with ML-ROS, ML-RUS and no-resampling algorithms; this is because ML-ROS/ML-RUS are better than LP-ROS/LP-RUS overall [Charte, Rivera, María del Jesus et al. (2015)]. At the same time, two different combination strategies of base classifiers and ensemble classifiers were compared with BR, RAKEL and CC. In addition, a 10-fold cross-validation strategy was adopted in the experiments.

**Parameter settings:** The over-sampling and under-sampling rates of ML-ROS and ML-RUS were all set to 0.1. The upper-sampling rate of RCS was 0.2, while the down-sampling rate was 0.05; moreover, the proportion of the upper-sampling was 0.3, while the proportion of the down-sampling was 0.1.

**Evaluation:** The performance of a specific multi-label classifier can be evaluated using a large range of measures [Wang, Kong, Li et al. (2019)], which can be broadly divided into two categories: macro-measures and micro-measures. There are several measures for each group, namely precision, recall and F1 score.

**Results and analysis:** To facilitate better analysis of the performance of the three sampling algorithms and two base classifier combination strategies, the experimental results are presented below in the form of line charts (see Fig. 3 to Fig. 8). In the figures, ‘rcs’ indicates the resampling algorithm proposed in this paper, ‘nos’ represents the no-resampling algorithm, ‘rus’ denotes the ML-RUS algorithm, and ‘ros’ represents the ML-ROS algorithm.

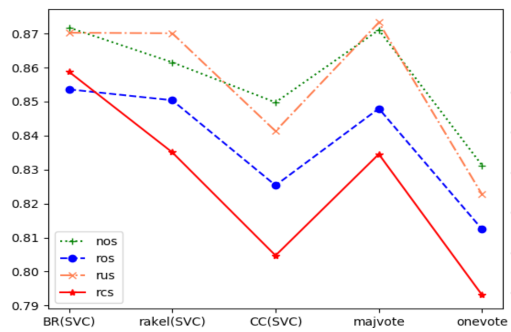


**Figure 3:** results-precision_micro

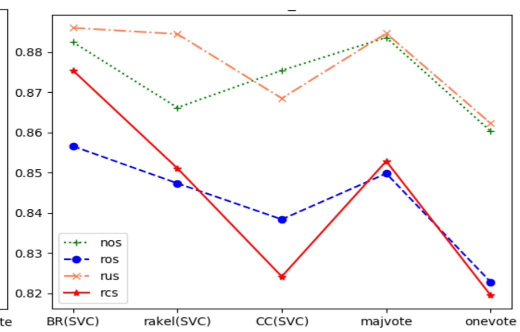


**Figure 4:** results-precision_macro

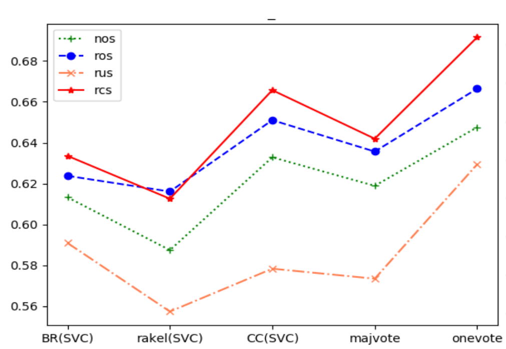


**Figure 5:** results-recall_micro

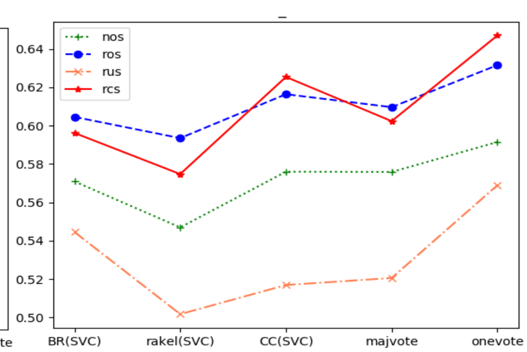


**Figure 6:** results-recall_macro

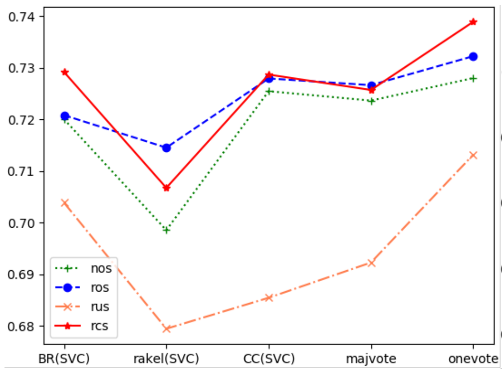


**Figure 7:** results-F1_micro

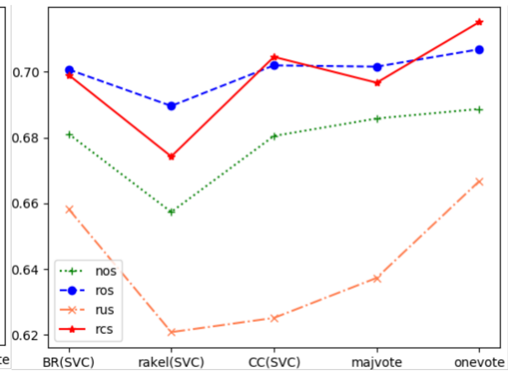


**Figure 8:** results-F1_macro

As can be seen from the above figures:

(1) For precision_micro and precision_macro, the performance of the ensemble classifier using the majority voting strategy is worse than that of the base classifiers, with the ensemble classifier employing the one-vote decision strategy yielding the worst performance. In terms of the sampling algorithm, ML-ROS and RCS show no significant advantages, while ML-RUS that performs slightly better than no-resampling.

(2) For recall_micro and recall_macro, the performance of the ensemble classifier using the majority voting strategy is generally better, or slightly worse, than the optimal performance of the base classifier, while the ensemble classifier using the one-vote decision strategy outperforms all base classifiers. Compared with the no-resampling strategy, ML-ROS and RCS resampling strategies can significantly improve recall, while the recall of ML-RUS decreases.

(3) For F1_micro and F1_macro, the results are similar to those of recall_micro and recall_macro. The main reason for this is that the increase in recall is higher than the decrease in precision, which leads to a significant increase of the F1 value.

In summary, the one-vote decision combination strategy achieves greater improvement when compared with the majority voting strategy. ML-ROS and RCS can significantly improve recall and F1 value, albeit by sacrificing precision to a certain extent, while ML-RUS is different. Finally, RCS-Bagging appears to achieve the best performance, which clearly demonstrates the effectiveness of the proposed model for the MLC context.

## 6 Conclusion

An MLC method based on key sentence extraction is proposed for MDD applications in this paper. Firstly, the method of extracting key sentences from MDDs can filter out a large amount of the irrelevant information contained in the documents and enable the classifiers to easily grasp the theme. Secondly, the method of combining resampling and Bagging can classify the key sentence sets, thereby enhancing the ability of the base classifiers to deal with class imbalance and further improving the classification performance. Experiments show that the proposed method substantially improves the effect of MLC on the original MDD dataset.

**Funding Statement:** This work was partly supported by the National Key R&D Program of China (2018YFC0830200, Zhang, B, www.most.gov.cn), the Fundamental Research Funds for the Central Universities (2242018S30021 and 2242017S30023, Zhou S, www.seu.edu.cn) and the Open Research Fund from Key Laboratory of Computer Network and Information Integration In Southeast University, Ministry of Education, China (3209012001C3, Zhang B, www.seu.edu.cn).

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

**Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M.** (2015): Internet of things: a survey on enabling technologies, protocols and applications. *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347-2376.

**Bahdanau, D.; Cho, K.; Bengio, Y.** (2014): Neural machine translation by jointly learning to align and translate. https://arxiv.org/abs/1409.0473.

**Breiman, L.** (1996): Bagging predictors. *Machine Learning*, vol. 24, no. 2, pp. 123-140.

**Cao, Z.; Wei, F.; Li, S.; Li, W.; Wang, H.** (2015): Learning summary prior representation for extractive summarization. *Proceedings of Meeting of the Association for Computational Linguistics & International Joint Conference on Natural Language Processing*, vol. 2, pp. 829-833.

**Charte, F.; Rivera, A. J.; María del Jesus, J.; Herreraac, F.** (2015): Addressing imbalance in multilabel classification: measures and random resampling algorithms. *Neuro Computing*, vol. 163, pp. 3-16.

**Cheng, J.; Lapata, M.** (2016): Neural summarization by extracting sentences and words. arXiv preprint arXiv:1603.07252.

**Gharaibeh, A.; Salahuddin, M. A.; Hussini, S. J.; Khreishah, A.; Khalil, I. et al.** (2017): Smart cities: a survey on data management, security and enabling technologies. *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2456-2501.

**He, D.; Ye, R.; Chan, S.; Guizani, M.; Xu, Y.** (2018): Privacy in the internet of things for smart healthcare. *IEEE Communications Magazine*, vol. 56, no. 4, pp. 38-44.

**He, S; Xie, K.; Xie, K.; Xu, C.; Wang, J.** (2019): Interference-aware multisource transmission in multiradio and multichannel wireless network. *IEEE Systems Journal*, vol. 13, no. 3, pp. 2507-2518.

**Hochreiter, S.; Schmidhuber, J.** (1997): Long short-term memory. *Neural Computation*, vol. 9, no. 8, pp. 1735-1780.

**Kupiec, J.; Pedersen, J.; Chen, F.** (1995): A trainable document summarizer. *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 68-73.

**Kurata, G.; Xiang, B.; Zhou, B.** (2016): Improved neural network-based multi-label classification with better initialization leveraging label co-occurrence. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 521-526.

**Nallapati, R.; Zhai, F.; Zhou, B.** (2017). SummaRuNNer: a recurrent neural network based sequence model for extractive summarization of documents. *Proceedings of the thirty-first AAAI conference on artificial intelligence*, pp. 3075-3081.

**Nam, J.; Kim, J.; Mencía, E. L.; Gurevych, I.; Fürnkranz, J.** (2014): Large-scale multi-label text classification-Revisiting neural networks. *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 437-452.

**Narayan, S.; Cohen, S. B.; Lapata, M.** (2018): Ranking sentences for extractive summarization with reinforcement learning. https://arxiv.org/abs/1802.08636.

**Sahi, M. A.; Abbas, H.; Saleem, K.; Yang, X.; Derhab, A. et al.** (2018): Privacy preservation in e-health care environments: State of the art and future directions. *IEEE Access*, vol. 6, pp. 464-478.

**Saravanan, M.; Ravindran, B.; Raman, S.** (2006): Improving legal document summarization using graphical models. *Frontiers in Artificial Intelligence and Applications*, vol. 152, pp. 51-60.

**Wang, B.; Kong, W.; Guan, H.; Xiong, N.** (2019): Air quality forecasting based on gated recurrent long short term memory model in internet of things. *IEEE Access*, vol. 7, pp. 69524-69534.

**Wang, B.; Kong, W.; Li, W.; Xiong, N. N.** (2019): A dual-chaining watermark scheme for data integrity protection in internet of things. *Computers, Materials & Continua*, vol. 58, no. 3, pp. 679-695.

**Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A. et al.** (2016): Hierarchical attention networks for document classification. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480-1489.

**Zhang M.; Zhou Z.** (2014): A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, pp. 1819-1837.