Combination of Loss-based Active Learning and Semi-supervised Learning for Recognizing Entities in Chinese Electronic Medical Records

JINGHUI YAN, School of Computer Science and Information Technology, Beijing Jiaotong University, China
CHENGQING ZONG, School of Computer Science and Information Technology, Beijing Jiaotong University, National Laboratory of Pattern Recognition, Institute of Automation Chinese Academy of Sciences, China
JINAN XU, School of Computer Science and Information Technology, Beijing Jiaotong University, China

The recognition of entities in an electronic medical record (EMR) is especially important to downstream tasks, such as clinical entity normalization and medical dialogue understanding. However, in the medical professional field, training a high-quality named entity recognition system always requires large-scale annotated datasets, which are highly expensive to obtain. In this article, to lower the cost of data annotation and maximizing the use of unlabeled data, we propose a hybrid approach to recognizing the entities in Chinese electronic medical record, which is in combination of loss-based active learning and semi-supervised learning. Specifically, we adopted a dynamic balance strategy to dynamically balance the minimum loss predicted by a named entity recognition decoder and a loss prediction module at different stages in the process. Experimental results demonstrated our proposed framework's effectiveness and efficiency, achieving higher performances than existing approaches on Chinese EMR entity recognition datasets under limited labeling resources.

CCS Concepts: • **Computing methodologies → Information extraction;**

Additional Key Words and Phrases: Electronic medical record, loss-based active learning, dynamic balance strategy, semi-supervised learning

ACM Reference format:
Jinghui Yan, Chengqing Zong, and Jinan Xu. 2023. Combination of Loss-based Active Learning and Semi-supervised Learning for Recognizing Entities in Chinese Electronic Medical Records. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 22, 5, Article 123 (May 2023), 19 pages.
https://doi.org/10.1145/3588314

Authors' addresses: J. Yan and J. Xu, School of Computer Science and Information Technology, Beijing Jiaotong University, Beijing, P.R. China, 100044; emails: {jh_yan, jaxu}@bjtu.edu.cn; C. Zong, School of Computer Science and Information Technology, Beijing Jiaotong University, Beijing 100049, P.R.China, 100044, and National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing 100049, P.R.China; email: cqzong@nlpr.ia.ac.cn.

2375-4699/2023/05-ART123 $15.00
https://doi.org/10.1145/3588314

1 INTRODUCTION

An **electrical medical record (EMR)** is a digital collection of health information about patients and populations [10]. It contains the diagnosis and treatment history of the patients, which are written in the form of medical text. EMR plays a critical role in medical applications, such as healthcare delivery and smart medicine [6, 7, 23]. Most of the text recorded in EMRs is unstructured. Therefore, automatically extracting information from EMRs plays an important role to downstream tasks such as EMR entity normalization [46] and medical dialogue understanding [25]. To support this process, medical **named entity recognition (NER)**, which is employed to extract pre-defined medical entities (e.g., diseases, diagnoses, examinations, and drugs), is the first and crucial step (as illustrated in Figure 1).

Recent advances in the applications of deep neural networks have inspired the development of NER systems [2, 8, 20]. Deep learning model success highly depends on the massive collection of annotated data. However, in EMR-related fields, tag acquisition is always expensive and time-consuming, which requires trained experts with domain knowledge [45]. Therefore, the study of methods to effectively annotate EMR data and build high-quality NER models with limited training data is extensive [24, 27, 30, 31, 44]. **Active learning (AL)** [3], which aims at interactively querying the most informative unlabeled data with limited budgets and try to train high-performance models with fewer queries, is a common method to address this issue.

Many AL approaches have been proposed to reduce the amount of data annotation [9, 22, 36, 37, 41]. The existing AL approaches can be mainly grouped into two categories: uncertainty sampling and diversity sampling. Uncertainty sampling uses confidence [22] and information entropy [14, 33, 39] to select informative samples. Diversity sampling selects diverse samples to represent the whole distribution of the unlabeled feature space [37, 48]. Regardless of which acquisition function is used, the core idea of AL is always to prioritize the most informative unlabeled samples to be selected for training that would lead to the greatest improvement in model performance. Reference [49] proposed a novel architecture to measure the informativeness of data points. It adopted a loss prediction module attached to the target network to predict the score of the loss function of data points. The assumption of the loss-based AL method is that data points that have high losses would be more informative to the training model.

Loss-based methods have achieved promising results in classification tasks but still have limitations. Table 1 shows an example of predicted losses and ground-truth losses of several data points. The predicted loss is provided by loss prediction module, and the ground-truth loss is directly calculated by target NER network after training iteration. Notice that the difference between the predicted value and the actual value is quite large. The "loss" predicted by the loss prediction module seems to be more of a sorting score than a training loss. Our empirical study indicated that this sorting score is less reliable than uncertainty methods as the number of training rounds increases (as demonstrated below). In this article, we aim at lowering the cost of data annotation of the Chinese EMR NER task. We find that a single sampling strategy has difficulty adapting to the entire model training process. Therefore, we propose a hybrid loss-based AL method to reduce the annotated cost. It considers the minimum loss directly predicted by the task NER model decoder and the loss predicted by the loss prediction module. We adopted a dynamic balance strategy to integrate these losses considering the tradeoff between the two components at different stages in the process of AL. Furthermore, we propose a new framework integrating **loss-based active learning strategies and semi-supervised learning (LASL)** to enhance the model performance by maximizing the use of unlabeled data. We compare our approach with existing methods [15, 22, 49], which have the best performance so far. We conducted extensive experiments to verify the effectiveness of our proposed models on the published Chinese EMR corpus. Experimental results demonstrated that our model can outperform strong baselines.

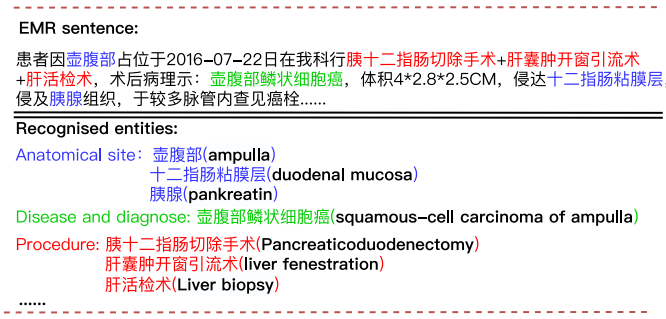

Fig. 1. An example of Chinese electrical medical record named entity recognition.

Table 1. Examples of Loss Values Predicted by Loss Prediction Module and Corresponding Ground-Truth Loss

Data sample	Predicted loss	Ground-truth loss
1	−9965.5419921875	0.116943359375
2	−10058.3427734375	0.02032470703125
3	−9418.607421875	0.18402099609375
4	7261.44580078125	0.047119140625
5	7959.1650390625	0.16778564453125
6	7552.912109375	0.3541259765625
7	7219.77685546875	0.03875732421875
...	...	...

In summary, the contributions of this article are threefold:

- A new hybrid loss-based AL method integrating multiple sampling strategies into the AL process has been proposed.
- A novel framework that integrates both LASL for training a Chinese EMR NER model has been developed, which can maximize the use of unlabeled data.
- The extensive experiments and ablation studies on Chinese EMR NER tasks have been conducted to evaluate the effectiveness of our LASL framework. Experimental results showed that our LASL framework achieved state-of-the-art performances under limited labeling resources.

The rest of this article is organized as follows. Section 2 presents the related work of AL and semi-supervised learning. Section 3 introduces our proposed LASL framework. The experimental setup is stated in Section 4, and the experimental results are shown in Section 5. Section 6 presents the conclusions.

2 RELATED WORK

We present the related work in this section in three sub-areas.

2.1 Named Entity Recognition

NER is one of the fundamental tasks within natural language processing and plays a critical role in medical information extraction applications. The task for EMRs is to take unstructured textual medical records and extract key information that is needed for downstream tasks. Earlier researchers adopted statistical methods such as support vector machines [17, 18, 21], hidden Markov

models [50, 52, 53], and conditional random fields [28, 38]. In recent years, deep learning methods have been adopted for NER tasks and have achieved promising results. Reference [5] first adopted a convolutional neural network as the encoder. Reference [16] proposed a **bidirectional long short-term memory (Bi-LSTM)** encoder combined with a **conditional random field (CRF)** decoder to solve the problem of long distance dependency. Refernece [20] shows the effectiveness of word representations in a Bi-LSTM-CRF architecture. The development of language models has also driven the iterative updating of NER models, such as the Embeddings from Language Models [40], transformer [26], and **bidirectional encoder representations from transformers (BERT)** [13] models. Reference [42] proposed a span-based NER model that leverages the relative positions through a multiplicative attention mechanism and achieves competitive results on both flat and nested NER datasets. In this article, we focuses on proposing an integrated framework with AL and semi-supervised learning to lower the cost of data annotation and maximizing the use of unlabeled data, and we evaluate the generalizability of our approach with different NER models in the experimental section of this article.

2.2 Active Learning

Active learning aims at reducing the labeling costs by iteratively querying oracles to annotate the most informative data from a large pool of unlabeled data. Various acquisition functions have been recently proposed for AL. The uncertainty approaches select data points by measuring the quantity of uncertainty. Least confidence [3] and max entropy [39] are two widely used uncertainty-based approaches. The diversity approaches [11, 29] select a set of unlabeled data that best represents the distribution of the dataset. Reference [37] defined diversity AL as a core-set selection problem based on the core-set distance of intermediate features. The loss-based approach is one of the key sources of inspiration for our work. Reference [49] proposed a task-agnostic loss prediction system for image classification tasks. By attaching a loss prediction module to the target classifier model, it was jointly trained to predict the loss for unlabeled data. The work of Reference [49] was only verified in the classification task dataset, while our empirical study for Chinese EMR NER datasets indicated that it appears to be less reliable as the target NER model's capabilities improve.

2.3 Semi-supervised Learning

Semi-supervised learning (SSL) is also an area related to our work. A variety of SSL methods work to address the lack of training data in information extraction. References [1, 4, 12, 47] adopted bootstrapping methods to learn from several seed examples and iteratively added the labeled portion of the data during the training process. More recently, teacher–student models [34, 35, 43] have been widely used in classification tasks. Reference [43] proposed the **mean teacher (MT)** framework, which learns by maximizing the consistency between a student model and a teacher model, producing impressive results for image classification. Reference [15] adopted the MT scheme for their proposed **temporal output discrepancy (TOD)** unlabeled data sampling strategy and achieved state-of-the-art active performances on image classification tasks. Inspired by Reference [15], we designed the MT architecture to be employed in NER tasks in conjunction with our proposed AL method to boost the task model performance with unlabeled data. The previous research did not integrate AL and SSL for training NER networks, In this work, we study the combination of AL and SSL for the Chinese EMR NER task.

3 INTEGRATED FRAMEWORK WITH ACTIVE LEARNING AND SEMI-SUPERVISED LEARNING

To reduce the annotation cost, we propose a new framework that integrates active learning and semi-supervised learning to improve the performance of the selected target NER model. The

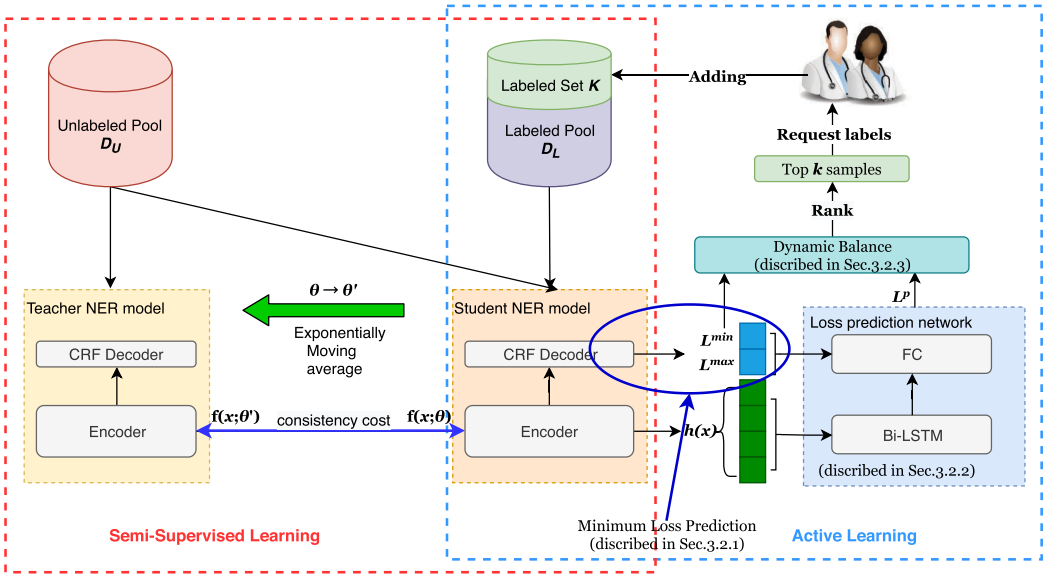

Fig. 2. Overview of our proposed integrated framework with active learning and semi-supervised learning. The blue dashed portion represents our loss-prediction-based active learning method (detailed in Section 3.2). The red dashed portion indicates the Mean Teacher–based semi-supervised learning method (detailed in Section 3.3). We adopted the Encoder-CRF architecture as our NER model (detailed in Section 3.1).

whole workflow of our proposed framework is depicted in Figure 2. There are two datasets in the workflow of the framework: the unlabelled dataset D_U and the labelled dataset D_L. We denote the initial NER model as M_0, which was pre-trained on the initial labelled dataset D_L^0. At the tth iteration, we denote the labeled dataset as D_L^t and the unlabeled pool as D_U^t. In each iteration, D_L^t is dynamically updated by the top k confidence samples from D_U^t with the loss-predication-based AL method (described in Section 3.2). Meanwhile, a semi-supervised learning method based on Mean Teacher architecture is employed to reduce the labeling costs (described in Section 3.3).

3.1 Target Named Entity Recognition Model

In this work, we adopted a character-based Encoder-CRF architecture as our target NER model. The architecture of our NER model consisted of two parts: (1) the feature encoder, which adopts character representation as input, was composed of a pre-trained language model and the Bi-LSTM layer and (2) the linear chain CRF layer was adopted as a decoder (shown in Figure 2 as the "student NER model"). Given an input sequence $\mathbf{x} = \{x_0, x_1, x_2, \ldots, x_n\}$, the NER model is trained to predict a vector $\mathbf{y} = \{y_0, y_1, y_2, \ldots, y_n\}$ of tags. For each correct sequence of tags, a conditional probability is computed,

$$p(\mathbf{y}|\mathbf{x}) = \frac{e^{Score(\mathbf{x},\mathbf{y})}}{\sum_{\mathbf{y}' \in Y_x} e^{Score(\mathbf{x},\mathbf{y}')}}, \tag{1}$$

where Y_x denotes all the sequence of tags, and $Score$ is computed as follows:

$$Score(\mathbf{x}, \mathbf{y}) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=i}^{n} P_{i, y_i}, \tag{2}$$

where $A_{y_i, y_{i+1}}$ denotes the transition probability, which represents the score of the transition from tag_i to tag tag_j, and $P_{i,j}$ is the non-normalized transition probability output by the encoder layer,

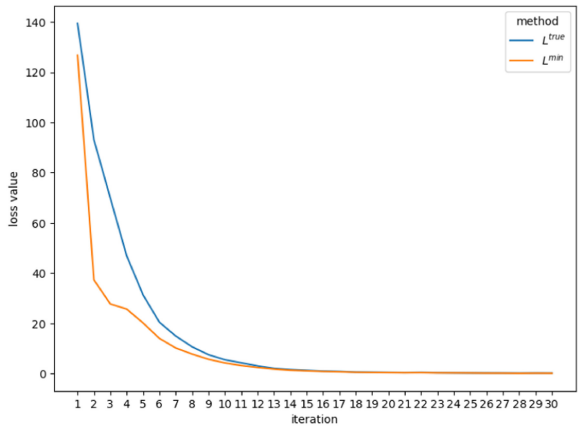

Fig. 3. Trends of minimum loss L^{min} and true loss L^{true} during training iterations.

which represents the score of tag_j of word w_i. Thus, in the training stage, we only need to maximize the likelihood probability $p(\mathbf{y}|\mathbf{x})$, and here we use $-log(p(y|x))$ as the loss function:

$$\begin{aligned} loss(\mathbf{x}, \mathbf{y}) &= -log\left(\frac{e^{Score(\mathbf{x},\mathbf{y})}}{\sum_{\mathbf{y}' \in Y_x} e^{Score(\mathbf{x},\mathbf{y}')}}\right) \\ &= \sum_{\mathbf{y}' \in Y_x} e^{Score(\mathbf{x},\mathbf{y}')} - Score(\mathbf{x}, \mathbf{y}). \end{aligned} \tag{3}$$

We use the "log-sum-exp trick" to sum a series of $e^{Score(\mathbf{x},\mathbf{y}')}$ terms. In the following sections, we use $N(\mathbf{x})$ to refer to $\sum_{\mathbf{y}' \in Y_x} e^{Score(\mathbf{x},\mathbf{y}')}$ for simplicity.

3.2 Loss-Prediction-based Active Learning Method

Given a pool of data, the basic idea of AL is to prioritize the data to be trained that would lead to the greatest improvement in model performance. The assumption of the loss-based AL method is that the data points that have high losses would be more informative to the training model. Based on this hypothesis, we propose two approaches to predict the data loss.

3.2.1 Minimum Loss Prediction. As shown in Equation (3), the loss function is composed of $N(\mathbf{x})$ and $e^{Score(\mathbf{x},\mathbf{y})}$. Using the log-sum-exp trick, the $N(\mathbf{x})$ can be obtained by direct calculation of the results. The problem is determining how to measure $Score(\mathbf{x}, \mathbf{y})$ when the label of the sample is unavailable. The Viterbi algorithm is adopted in this work to find the entire sequences $\mathbf{y}_{max}$ and $\mathbf{y}_{min}$ such that the sum of the transition scores $Score(\mathbf{x}, \mathbf{y})$ is maximized and minimized, respectively. Then, for each given sequence $\mathbf{x}$, we can estimate the limit of loss as follows:

$$\begin{aligned} L^{true}(\mathbf{x}, \mathbf{y}) > L^{min} &= N(\mathbf{x}) - Score(\mathbf{x}, \mathbf{y}^{max}), \\ L^{true}(\mathbf{x}, \mathbf{y}) < L^{max} &= N(\mathbf{x}) - Score(\mathbf{x}, \mathbf{y}^{min}). \end{aligned} \tag{4}$$

Figure 3 depicts the loss value trends for the minimum loss $L^{min}(\mathbf{x})$ and true loss $L^{true}(\mathbf{x}, \mathbf{y})$. $L^{min}(\mathbf{x})$ approaches $L^{true}(\mathbf{x}, \mathbf{y})$ as the number of training data increases. This demonstrates that given an input sequence $\mathbf{x}$ and ground-truth label $\mathbf{y}$, the calculated L^{min} can be more and more regarded as an approximation of L^{true} as the target NER model improves. However, in the early training stage, the lack of training data results in a lower effectiveness of the emission probability and the transition probability mentioned in Equation (2). Thus, the sequence $\mathbf{y}^{max}$ obtained at early training stage is unreliable.

3.2.2 Loss Prediction Network. Inspired by Reference [49], we use a regression network to dynamically predict the current loss of the input unlabeled data. In our design, the loss prediction network is attached to the target NER model and uses the Bi-LSTM hidden layer of the target NER model to obtain the feature $h(x)$ as an input. As illustrated in the right side of Figure 2, the loss prediction network includes a Bi-LSTM layer and a **fully connected (FC)** layer. Here L^{min} and L^{max} mentioned in Section 3.2.1 are employed as two feature values to concatenate the feature vectors from the Bi-LSTM and pass through a FC layer.

In the training process, the NER loss is computed by comparing the predicted probability and ground-truth labels with certain criteria. The loss prediction network takes the target NER loss as the ground-truth labels for training. Note that the scale of the true loss change substantially during the learning of the target NER network. To discard the impact of overall scale changes, A data-pair-based loss function is defined instead of minimizing mean square error. Consider a mini-batch with size B in a training iteration, we make B data pairs (x_i, x_j), where $j = \begin{cases} 1 & i+1 > B \\ i+1 & otherwise \end{cases}$. Compared with Reference [49] that limits B to an even number and treats each batch as $B/2$ data pairs, our way of pairing has no restrictions on the number of batch size. Then, the loss prediction module can be trained by comparing a pair of loss predictions. The loss function is designed as follows:

$$\begin{aligned} L(\hat{l}^p, l^p) &= max\left(0, -E(l_i, l_j)(\hat{l}_i - \hat{l}_j) + \epsilon\right) \\ E(l_i, l_j) &= \begin{cases} 1 & l_i > l_j \\ -1 & otherwise \end{cases}, \end{aligned} \tag{5}$$

where l^p represents the loss pair (l_i, l_j), $\hat{l}^p$ represents the predicted loss pair $(\hat{l}_i, \hat{l}_j)$, and ϵ is a positive margin constant. When $l_i > l_j$, the expected module should give the prediction $\hat{l}_i > \hat{l}_j + \epsilon$; otherwise, the $\hat{l}_i$ and $\hat{l}_j$ should be increased and decreased respectively by the loss prediction module.

3.2.3 Dynamic Balance Strategy. As discussed before, we use two approaches to measure the loss of each data point in the unlabeled subset. We propose a dynamic balance strategy to integrate the minimum loss prediction and loss prediction network and consider the tradeoff between the two methods at different phases in the process of the AL algorithm. In the early stage of AL, the lack of training data results in unreliable results for the target model decoder. At this moment, the selected data should be dependent more on the *loss prediction network*, which is less constrained by the decoder. After several iterations, L^{min} becomes increasingly close to the true loss as the model's decoder ability increases. At this moment, we select data based more on the *minimum loss prediction* to better improve the target model. Given a target NER model M, for each instance x_i at learning cycle t, an intuitive way to integrate the two approaches is to define a linear combination as follows:

$$rank(x_i) = \Lambda_t \times Softmax\left(L_i^{min}\right) + (1 - \Lambda_t) \times Softmax\left(L_i^p\right), \tag{6}$$

where L_i^{min} and L_i^p are the loss values predicted by the *minimum loss prediction* and *loss prediction network* of x_i, respectively. Consider that different types of loss metrics have different data scales, we use softmax to map the values of the two predicted losses to the same scales. Λ_t is a weight parameter that represents the ability of the task model decoder at learning cycle t. In this article, we use the mean conditional probability to measure the competence of the model decoder,

$$\Lambda_t = \frac{1}{|D_U^t|} \sum_{i \in D_U^t} P\left(y_i = \mathbf{y}^{max} | x_i; M^t\right), \tag{7}$$

where M^t_{task} represents the task NER model at iteration t and $P(y_i = \mathbf{y}^{max}|x_i; M^t_{task})$ is the probability of instance x_i belonging to $\mathbf{y}^{max}$ based on NER model M^t_{task}. Intuitively, as the learning iteration t increases, Λ_t will become larger and larger, which indicates a higher reliability of the decoding results, and informative samples will be selected depended from L^m_i to more on L^p_i.

3.3 Semi-supervised Task Learning

To maximize the use of unlabeled data, we adopted the Mean Teacher approach [43], which is a semi-supervised learning framework, for our task. This framework learns by maximizing the consistency between a student model and a teacher model that is an average of past students, where both classifiers are exposed to different noise. As illustrated in Figure 2 (red dotted block), the student model is our original NER model and is updated via backpropagation. For the teacher NER model, we apply an **exponential moving average (EMA)** to the latest version of the student parameters, as

$$\theta'_t = \alpha\theta'_{t-1} + (1-\alpha)\theta_t, \tag{8}$$

where θ'_t is defined at the training step t as the EMA of successive weights θ and α is the EMA decay rate. In each iteration t, we calculate the consistency cost for the unlabeled pool D^t_U to minimize the differences predictions between the teacher and student NER models. The consistency cost L^t_U is

$$L^t_U = \frac{1}{|D^t_U|}\sum_{x\in D^t_U} ||f(x;\theta) - f(x;\theta')||^2, \tag{9}$$

where $f(x;\theta)$ and $f(x;\theta')$ are the predictions of the student model with weights θ and the teacher model with weights θ', respectively. For simplicity, here the predictions are the non-normalized transition probability distributions output by our NER models ($P_{i,j}$ in Equation (2)). To integrate our NER, AL, and unsupervised losses, we minimize an overall learning objective as follows:

$$L_{overall} = L^t_{task} + L^t_{AL} + \lambda L^t_U, \tag{10}$$

where t is the AL iteration, L_{task} is the loss of the NER task calculated by Equation (3), L_{AL} is the loss of the loss prediction network calculated by Equation (5), L^t_U is the unsupervised loss calculated by Equation (9), and λ is a weight parameter to balance the task and unsupervised loss terms.

3.4 Algorithm Implementation

Our framework runs the LASL algorithm described in Algorithm 1. Initially, a small set of unlabeled data is selected randomly to be labelled as the initial labeled data for training all the initial models. At each iteration t, we first adopt our loss-based AL methods to select the most informative samples from D^{t-1}_U. For each instance in D^{t-1}_U, we calculate two loss scores according to the loss prediction network and the minimum loss prediction, which is based on the student NER model M^{t-1}_{task} and the loss prediction model M^{t-1}_{loss} at the last iteration, respectively, in lines 4 and 5. In line 6, we calculate the probability of instance x_i belonging to $\mathbf{y}^{max}$ based on model M^{t-1}_{task}, which is an operator to obtain the dynamic balance parameter in line 8 according to Equation (7). After this, we integrate the two loss scores and calculate the final *rank* score according to Equation (6). Then we select the top k samples based on the final *rank* score at each iteration. The selected sample set K is annotated by oracle and incorporated into the labeled dataset D^{t-1}_L. Finally, we jointly learn the

ALGORITHM 1: Algorithm for LASL

Input: D_L: labeled dataset
D_U: unlabeled dataset
M^0_{task}: initial student NER model
$M^0_{teacher}$: initial teacher NER model
M^0_{loss}: initial loss prediction model
k: number of selected data points in each iteration
T: maximum iteration number
Output: M^T_{task}: fine-tuned NER model at iteration t
Initialisation : $D^0_U \leftarrow D_U$, $D^0_L \leftarrow \emptyset$, $t \leftarrow 0$
while t not reach maximum round T **do**
  $t = t + 1$
  for all $x_i \in D^{(t-1)}_U$ **do**
    Calculate L^{min}_i according to *minimum loss prediction* using model $M^{(t-1)}_{loss}$
    Calculate L^p_i according to *loss prediction network* using model $M^{(t-1)}_{task}$
    Calculate $P(y_i = \mathbf{y}^{max}|x_i)$ using model $M^{(t-1)}_{task}$
  end for
  Calculate Λ_t according to Equation (7)
  Query top k samples K from D^t_U according to Equation (6)
  Label K by oracle
  $D^t_U \leftarrow D^{(t-1)}_U - K$ and $D^t_L \leftarrow D^{(t-1)}_L + K$.
  Calculate L^t_{task} according to Equation (3) using D^t_L
  Calculate L^t_{AL} according to Equation (5) using D^t_L
  Calculate L^t_U according to Equation (9) using D^t_U
  Fine-tune the models $M^{(t-1)}_{loss}$ and $M^{(t-1)}_{task}$ according to Equation (10) and obtain $M^{(t)}_{task}$ and $M^{(t)}_{task}$
  Update teacher model $M^{(t-1)}_{teacher}$ according to Equation (8) and obtain $M^{(t)}_{teacher}$
end while
return M^T_{task}

three loss functions of the task NER model, loss prediction model, and Mean Teacher. These steps are repeated until the maximum iteration number is reached.

4 EXPERIMENTS

4.1 Datasets and Evaluation Metrics

Dataset: We conducted experiments on the CCKS19-CNER dataset.[1] This is a Chinese EMR NER dataset provided by the **China conference on knowledge graph and semantic computing (CCKS)**. It includes a training set and a test set. The training set includes 1,000 medical records, manually labeled with six types of entities (including diseases, diagnoses, examinations, tests, procedures, drugs, and body parts). The test set contains 379 medical records of raw data.

Evaluation Metrics: We employed the F1-score as our evaluation metric of the task NER model. We evaluated the results of the NER model after each iteration of AL.

[1] https://www.biendata.xyz/competition/ccks_2019_1/.

Table 2. Main Hyper-parameters

Model	NER network	loss prediction network
Bi-LSTM layer	1	2
Bi-LSTM size	200	200
dropout	0.2	0.2
learning rate	2e-5	0.01
weight decay	1e-5	5e-4
momentum	—	0.9
batch size	16	16

4.2 Baseline

To demonstrate that our proposed LASL framework is more effective for NER tasks, we compared it with the following other methods:

- All Data: All the training samples are already labeled to train the task NER model.
- Random Sampling: Using this acquisition function is equivalent to choosing a point uniformly at random from the unlabeled data pool.
- Least-Confidence-Based Sampling: A common AL baseline that selects the most uncertain samples according to least confidence.
- Loss Sampling: Samples are selected according to the predicted loss by the vanilla loss prediction module proposed in Reference [49].
- TOD Sampling: A novel unlabeled data sampling proposed in Reference [15] using the temporal output discrepancy to measure the potential loss of a sample that only relies on the training model.

4.3 Implementation Details

To ensure the fairness of the experiment, all the methods adopted the same NER architecture described in Section 3.1. We adopt MC-BERT [51] as our pre-trained language model in encoder, which is pre-trained over Chinese medical corpora via masking different granularity tokens. The main hyper-parameters are described in Table 2. We adopted the Adam [19] optimizer and the SGD optimizer to train the NER models and loss prediction network, respectively. The initial NER models of all sampling methods were trained by the same random seed sample of 1% of the data in the training set. Following that, in each AL iteration, the AL methods chose an additional 1% of the unlabeled D_U samples to be labeled and added them to the training set until 30 iterations. For the semi-supervised, we set the EMA decay rate α to 0.999 and the tradeoff weight λ to 0.05, which are the same values used previously [15].

5 RESULTS AND ANALYSIS

In this section, we compare our proposed LASL framework with existing methods on experimental datasets. Table 3, Figure 4, and Figure 5 detail the results of our experiments. To evaluate the contributions of individual components to the LASL, we also experimented with model variants, where some of the components were removed or added. In more detail, loss sampling denotes the vanilla loss prediction module proposed in Reference [49]. **Minimum loss prediction (MLP)** denotes that only the minimum loss prediction module in the LASL (described in Section 3.2.1) was retained. **Loss prediction network (LPN)** denotes our modified loss prediction network, with two added feature vectors L^{min} and L^{max} (described in Section 3.2.2). Furthermore, DBS means that our dynamic balance strategy was adopted to integrate our MLP and LPN components

Table 3. F1-scores of the Active Learning Methods of Different Scenarios in CCKS19-CNER

Model	iteration (percentage of training set)					
	5(%)	10(%)	15(%)	20(%)	25(%)	30(%)
random	0.223	0.643	0.728	0.761	0.788	0.803
LC	0.449	0.734	0.796	0.813	0.819	0.823
TOD	0.368	0.731	0.777	0.804	0.812	0.824
loss sampling	0.484	0.735	0.768	0.788	0.805	0.810
MLP	0.449	0.740	0.794	0.813	0.821	0.822
LPN	0.485	0.752	0.792	0.802	0.813	0.815
DBS	0.478	0.766	0.804	0.814	0.824	0.827
LASL	0.508	0.771	0.806	0.819	0.826	0.832

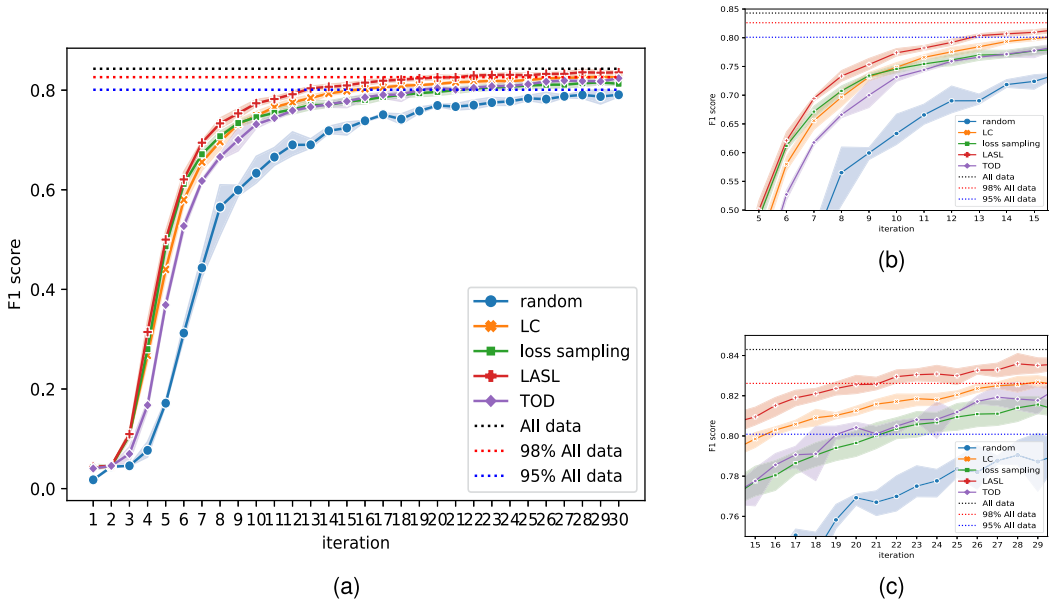

Fig. 4. Comparison of the proposed LASL with different methods based on the F1-score. Panels (b) and (c) show the zoomed-in portions of two different regions in (a). The shaded areas represent the standard deviation.

(described in Section 3.2.3). LASL denotes our proposed complete framework, including a Mean Teacher semi-supervised learning module.

5.1 Main Results

Table 3 shows the F1-scores of the AL methods with different AL iterations. We use the result of the all data (F1-score: 0.842) as a benchmark to evaluate the effectiveness of the AL methods. The numbers in blue and red in the background indicate that the methods in this iteration outperformed the 95% and 98% benchmarks, respectively. All the AL methods outperformed the random sampling

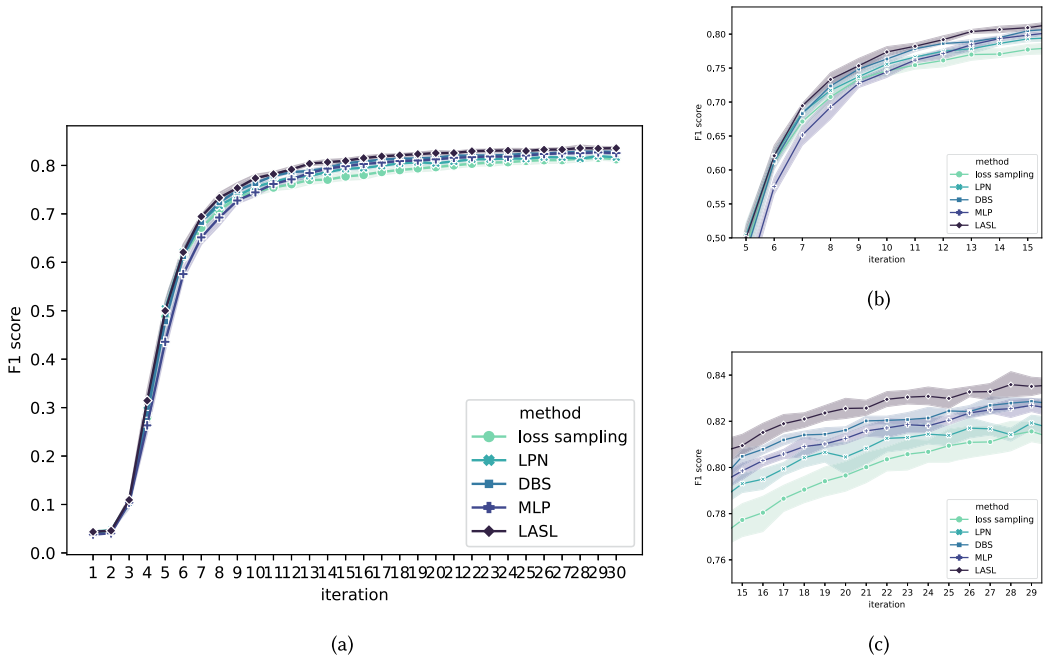

Fig. 5. Comparison of LASL with its variants using the F1-score. Panels (b) and (c) show zoomed-in portions of two different regions in (a). The shaded areas represent the standard deviation.

strategy and achieved more than the 95% upper limit within 30 training rounds. Our proposed LASL achieved better performances than all the other methods for the same number of training iterations and is the only method to reach the 98% upper limit within 25 iterations. Figure 4 illustrates the variations of the values for different AL methods. The blue dashed line at the top represents the All data benchmark. In the very early stages (1–5 iterations), our approach was slightly ahead of the loss sampling strategy. As the number of training rounds was increased, there was a sharp rise in the curve of our LASL, and it reached the 95% line using only 13 training iterations (13% of the training set), which was five and seven rounds earlier than the TOD and loss sampling strategy, respectively. At the 22nd training iteration, our method reached the 98% benchmark, which was at least eight iterations earlier than others. This means that our proposed LASL can effectively reduce the required labor of labelling samples in training deep learning models for Chinese medical NER tasks compared to other AL methods. In addition, it is worth noting that the performance of the loss sampling strategy was better than the LC strategy in the early iterations but was overtaken after 10 iterations. The reason may be that the calculation of least confidence had the same flaws as our proposed MLP, which is highly constrained by the task model decoder, while the lack of training data in the early stages resulted in unreliable results for the CRF decoder in the NER model.

Then, as illustrated in Figure 5, we compared the performances of the LASL variants to evaluate the contribution of each component of our method. The LPN outperformed the loss sampling (vanilla loss prediction module) at each iteration, especially when the amount of training data increased. This may indicate that the two feature values L^{min} and L^{max} could contribute to constraining the upper and lower bounds for the loss prediction. The comparison of the MLP with the LPN showed that the LPN was ahead of the MLP until 13 training iterations, after which the MLP curve clearly rose and surpassed the LPN. This illustrated that our proposed LPN and MLP sampling strategies had an edge in the early and late training periods, respectively. Meanwhile,

Table 4. F1-scores of LASL with Different Target NER Models

model	5%	10%	20%	30%
LSTMCRF	0.171 (0.523)	0.633 (0.766)	0.754 (0.815)	0.791 (0.824)
BERTCRF	0.223 (0.508)	0.643 (0.771)	0.761 (0.819)	0.803 (0.832)
Globalpointer	0.219 (0.499)	0.637 (0.743)	0.728 (0.765)	0.788 (0.794)

the DBS curve was a good combination of the LPN and MLP. This demonstrated that our proposed dynamic balance strategy could integrate the LPN and MLP and achieve a tradeoff between the two methods at different phases in the process of the AL algorithm. Furthermore, the LASL outperformed the DBS in all cases, which validated the effectiveness of our semi-supervised learning module for reducing the required labor for labelling Chinese EMR NER samples.

To further validate the generalizability of LASL, we separately test three different target NER models: (1) LSTMCRF, a basic sequence labeling method with Bi-LSTM as word encoder; (2) BERTCRF, the same model as described in Section 3.1; and (3) Globalpointer [42], a span-based NER model that leveraging the relative positions through a multiplicative attention mechanism. As shown in Table 4, we compared the performances of three target NER models using the random sampling and the LASL method (values in parentheses). The results show that the F1 scores of all these models using the LASL improves significantly faster than random sampling as the training data increases, which means that, with the same amount of annotated data, the target NER model with LASL sampling can always outperform the traditional method, especially when the labeled training samples are sparse (there are 0.352, 0.285, and 0.28 improvement separately with 5% labeled training data).

5.2 Study on Balance Strategy

5.2.1 Effect of Initial Training Data and Selecting Data. Based on the performance changes in our two proposed loss-based active methods, we determined their respective advantages in different training stages. To examine this further, we examined the MLP and LPN with different numbers of initial training data and selected data in each training iteration. The results are illustrated in Figure 6. We found that with certain amounts of initial training data, with more data selected at each iteration for the NER model, the MLP outperformed the LPN early. Figure 6(a) and (b) show the results with the same 1% sampled data from the original training set as the initial training data for task NER models. In Figure 6(a), 1% of the unlabeled D_U samples were labeled and trained at each training iteration, whereas, in Figure 6(b), 2% of the selected unlabeled samples were labeled and trained. We can see that MLP beyond LPN is moving up from the 13th (in Figure 6(a)) to the 5th (in Figure 6(b)) training interation. Furthermore, when the amount of selected data at each iteration was known, the more initial training data there was for the NER model, the earlier the MLP outperformed the LPN. In Figure 6(a), (c), and (d), the amount of selected data at each training iteration was the same, while the amounts of initial training data sampled from the original training set were 1%, 5%, and 10%, respectively. The F1-score of the MLP exceeded that of the LPN in these three figures at the 13th, 3rd, and 2nd training iteration, respectively. The above-mentioned facts confirmed that the lack of training data resulted in unreliable results of the target model decoder, and thus the LPN strategy, which was less constrained by the decoder, could provide better results than the MLP. More initial training data and data points selected resulted in a more robust decoder, and the model with a stronger decoder could provide an L^{min} that was closer to the true loss. That was the reason that we proposed a dynamic balance strategy to integrate the MLP and LPN that considered the tradeoff between the two methods at different training stages. From Figure 6(a) to (d), we can see that all the DBS curves could well integrate the LPN and MLP methods for all combinations of initial training data and selected data.

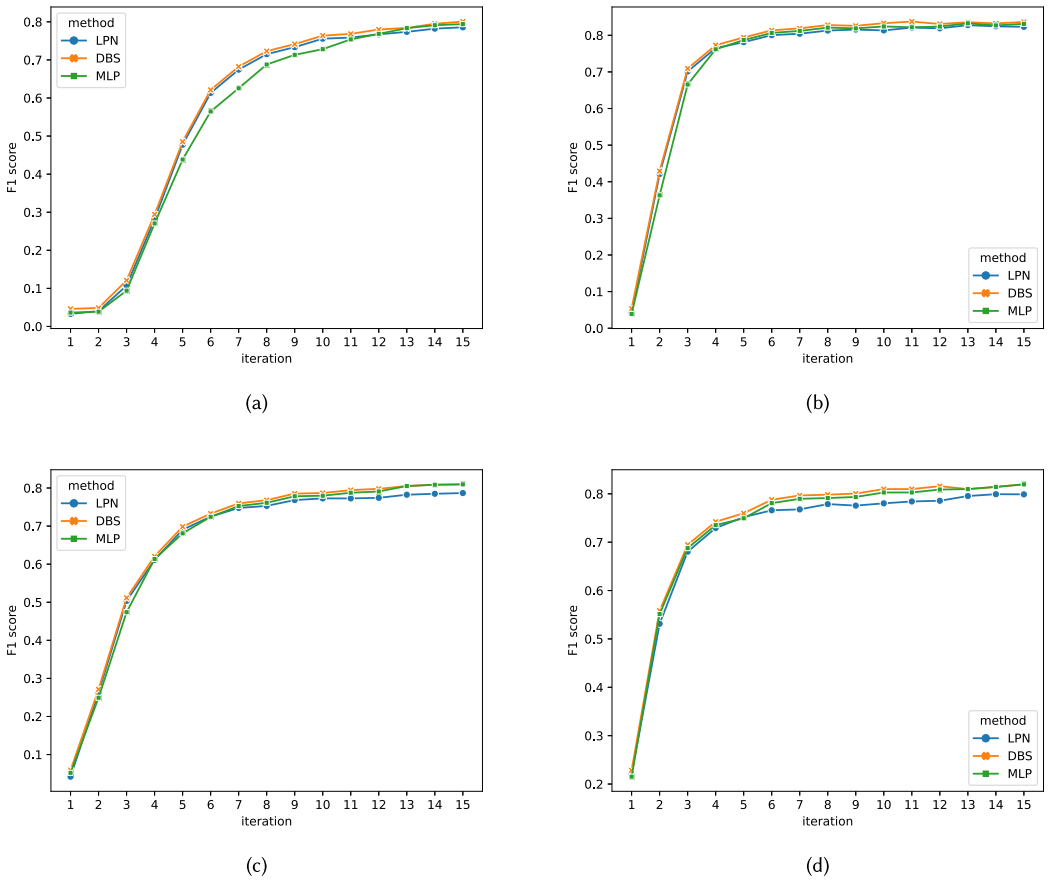

Fig. 6. Effectiveness of proposed dynamic balance strategy with different numbers of initial training data and selected data at each training iteration: (a) initial: 1%, selected: 1%; (b) initial: 1%, selected: 2%; (c) initial: 5%, selected: 1%; and (d) initial: 10%, selected: 1%.

5.2.2 Effect of Weight Parameter. Based on the performance changes in Figure 6, the use of the DBS module could integrate the LPN and MLP methods and improve the performance. In our DBS module, we employed the mean conditional probability as the weight parameter to dynamically measure the competence of the model decoder. To examine the effectiveness of our weight parameter, we conducted experiments with three different weight parameters as follows:

- Equal: Λ_t was directly set to 0.5, which means the LPN and MLP have the same weight during all the training iterations
- Linear: The weight parameter varied linearly with training iteration t:

$$\Lambda_t = t\frac{1-c_0}{T} + c_0 \tag{11}$$

- Root: Proposed in Reference [32], a root variant of Linear results in the following definition:

$$\Lambda_t = \sqrt[p]{t\frac{1-c_0^p}{T} + c_0^p}, \tag{12}$$

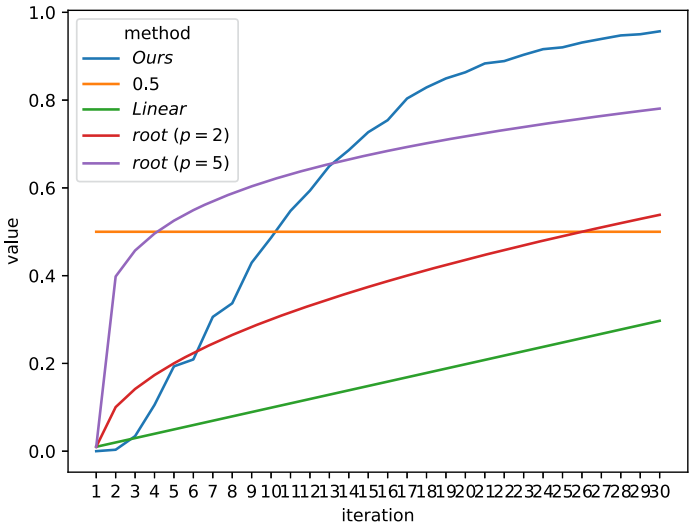

Fig. 7. Plots of various weight parameters during 30 active training iterations.

Table 5. The F1-scores of Dynamic Balance Strategies with Different Weights Parameters on CCKS19-CNER

Weight parameter	iteration					
	5	10	15	20	25	30
Equal	0.459	0.765	0.804	0.814	0.822	0.826
Linear	0.456	0.763	0.795	0.807	0.819	0.828
Root (p = 2)	0.454	0.758	0.791	0.810	0.819	0.829
Root (p = 5)	0.410	0.762	0.798	0.812	0.823	0.823
Ours	**0.478**	**0.766**	**0.804**	**0.814**	**0.824**	**0.827**

where c_0 is an initial value and T denotes the total number of iterations required to train all the data in the training set. We set $c_0 = 0.001$ and $T = 100$. Figure 7 illustrates the trends of various weight parameters mentioned above during 30 active training iterations (initial training data and selected data were the same as in Figure 6(a)). The trend of our proposed weight parameter was broadly in agreement with the trends of the MLP and DBS in Figure 5. We note that the value of the weight parameter exceeded 0.5 after the 10th iteration, which also coincided with the iteration when the MLP started to overtake the LPN in Figure 6(a). This demonstrated that our proposed mean conditional probability could better dynamically reflect the competence of a model than others without any manually set parameters. We tested all the weight parameters above in our DBS module. The results in the Table 5. demonstrate that the model performance based on our proposed mean conditional probability (highlighted in bold) is consistently better across all iteration rounds compared to other weight parameters.

5.3 Study on Encoder of Loss Prediction Network

Different from the method reported in Reference [49], which is a task-agnostic loss-based system for image classification, our LPN module was adapted for NER with domain-specific datasets. Here we further investigate the encoder selection for our proposed loss prediction network. We conducted experiments with different encoder structures, as shown in Figure 8. "Shared encoder"

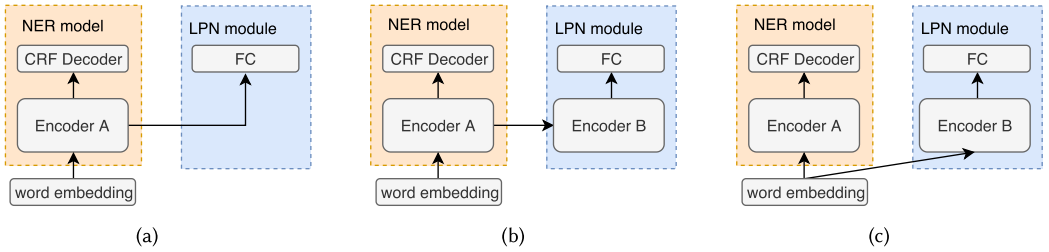

Fig. 8. Overview of different loss prediction network structures: (a) shared encoder, (b) serial encoder, and (c) individual encoder.

Table 6. The F1-scores of the Loss Prediction Network with Different Encoders on CCKS19-CNER

Structure		Iteration					
		5	10	15	20	25	30
Shared		0.401	0.698	0.734	0.790	0.799	0.807
Serial	1-layer	0.432	0.743	0.787	0.795	0.807	0.812
	2-layer	0.485	**0.752**	0.792	**0.802**	**0.813**	**0.815**
	3-layer	**0.487**	0.751	**0.789**	0.789	0.809	0.814
Individual	1-layer	0.382	0.653	0.721	0.746	0.789	0.798
	2-layer	0.421	0.699	0.767	0.785	0.792	0.802
	3-layer	0.454	0.713	0.767	0.782	0.792	0.802

means that LPN shared the same encoder with the task NER model. The output feature maps of each layer (if the encoder had multiple layers) in Encoder A were concatenated and directly transferred to the FC layer to output the loss prediction. "Serial encoder" means that the LPN adapted an additional Encoder B to re-encode the feature from Encoder A. "Individual encoder" means that the embedding input was encoded separately by two encoders. The results are shown in Table 6. For simplicity, we adopted MC-BERT [51] for Encoder A and Bi-LSTM for Encoder B. More sophisticated language model encoders are possible, but we did not consider them in this experiment. The number of layers of Encoder A was fixed, and the number of layers of Encoder B was variable. It can be concluded from the results that in most cases, the "Serial 2-layer" achieved the best performance. As a result, for our main experiment in this article, we employed a "Serial Encoder" with two Bi-LSTM layers.

6 CONCLUSION

In this work, we proposed a new framework integrating loss-based AL and semi-supervised learning for Chinese EMR NER to reduce the annotated cost. This framework contains two simple yet effective loss-based AL methods and a dynamic balance strategy to integrate them. The tradeoff between the two methods at different stages in the process of AL is considered. We evaluated our framework on a Chinese EMR NER dataset. Extensive experimental results demonstrated the merits of our proposed framework compared with the previous uncertainty-based and loss-based AL methods. We also conducted detailed analysis to illustrate the effects of different components in our framework. Furthermore, experiments demonstrate the good generalizability of our model, the components of the target NER network and loss prediction network are replaceable and easily extended by different models.

REFERENCES

[1] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, and Tom M. Mitchell. 2010. Toward an architecture for never-ending language learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 1306–1313.

[2] Jason P. C. Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional LSTM-CNNs. *Trans. Assoc. Comput. Ling.* 4 (2016), 357–370.

[3] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. 1996. Active learning with statistical models. *J. Artif. Intell. Res.* 4 (1996), 129–145.

[4] Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *Proceedings of the 1999 Joing SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)*. 100–110.

[5] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* 12, (2011), 2493–2537.

[6] Martin R. Cowie, Juuso I. Blomster, Lesley H. Curtis, Sylvie Duclaux, Ian Ford, Fleur Fritz, Samantha Goldman, Salim Janmohamed, Jörg Kreuzer, Mark Leenay, et al. 2017. Electronic health records to facilitate clinical research. *Clin. Res. Cardiol.* 106, 1 (2017), 1–9.

[7] Spiros C. Denaxas and Katherine I. Morley. 2015. Big biomedical data and cardiovascular disease research: Opportunities and challenges. *Eur. Heart J. Qual. Care Clin. Outcomes* 1, 1 (2015), 9–16.

[8] Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. Character-based LSTM-CRF with radical-level features for Chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications*. Springer, 239–250.

[9] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep bayesian active learning with image data. In *Proceedings of the International Conference on Machine Learning (ICML'17)*. PMLR, 1183–1192.

[10] Tracy D. Gunter and Nicolas P. Terry. 2005. The emergence of national electronic health record architectures in the United States and Australia: Models, costs, and questions. *J. Med. Internet Res.* 7, 1 (2005), e383.

[11] Yuhong Guo. 2010. Active instance sampling via matrix partition. In *Proceedings of the 23rd International Conference on Neural Information Processing Systems*. 802–810.

[12] Sonal Gupta and Christopher D. Manning. 2015. Distributed representations of words to guide bootstrapped entity classifiers. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1215–1220.

[13] Kai Hakala and Sampo Pyysalo. 2019. Biomedical named entity recognition with multilingual BERT. In *Proceedings of the 5th Workshop on BioNLP Open Shared Tasks*. 56–61.

[14] Alex Holub, Pietro Perona, and Michael C. Burl. 2008. Entropy-based active learning for object recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 1–8.

[15] Siyu Huang, Tianyang Wang, Haoyi Xiong, Jun Huan, and Dejing Dou. 2021. Semi-supervised active learning with temporal output discrepancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3447–3456.

[16] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. arXiv:1508.01991. Retrieved from https://arxiv.org/abs/1508.01991.

[17] Hideki Isozaki and Hideto Kazawa. 2002. Efficient support vector classifiers for named entity recognition. In *Proceedings of the 19th International Conference on Computational Linguistics*. 1–7.

[18] Zhenfei Ju, Jian Wang, and Fei Zhu. 2011. Named entity recognition from biomedical text using SVM. In *Proceedings of the 5th International Conference on Bioinformatics and Biomedical Engineering*. IEEE, 1–4.

[19] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

[20] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 260–270.

[21] Ki-Joong Lee, Young-Sook Hwang, Seonho Kim, and Hae-Chang Rim. 2004. Biomedical named entity recognition using two-phase model based on SVMs. *J. Biomed. Inf.* 37, 6 (2004), 436–447.

[22] David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94)*. Springer, 3–12.

[23] Irene Li, Jessica Pan, Jeremy Goldwasser, Neha Verma, Wai Pan Wong, Muhammed Yavuz Nuzumlalı, Benjamin Rosand, Yixin Li, Matthew Zhang, David Chang, et al. 2021. Neural natural language processing for unstructured data in electronic health records: A review. arXiv:2107.02975. Retrieved from https://arxiv.org/abs/2107.02975.

[24] Muqun Li, Martin Scaiano, Khaled El Emam, and Bradley A. Malin. 2019. Efficient active learning for electronic medical record de-identification. In *AMIA Summits on Translational Science Proceedings*, 462.

[25] Mei Li, Lu Xiang, Xiaomian Kang, Yang Zhao, Yu Zhou, and Chengqing Zong. 2021. Medical term and status generation from chinese clinical dialogue with multi-granularity transformer. *IEEE/ACM Trans. Audio Speech Lang. Process.* 29 (2021), 3362–3374.

[26] Xiaonan Li, Hang Yan, Xipeng Qiu, and Xuan-Jing Huang. 2020. FLAT: Chinese NER using flat-lattice transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 6836–6842.

[27] Qunsheng Ma, Xingxing Cen, Junyi Yuan, and Xumin Hou. 2021. Word embedding bootstrapped deep active learning method to information extraction on Chinese electronic medical record. *J. Shanghai Jiaotong Univ. (Sci.)* 26, 4 (2021), 494–502.

[28] Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the 7th conference on Natural language learning at HLT-NAACL*. 188–191.

[29] Hieu T. Nguyen and Arnold Smeulders. 2004. Active learning using pre-clustering. In *Proceedings of the 21st International Conference on Machine Learning*. 79.

[30] Minh-Tien Nguyen, Guido Zuccon, Gianluca Demartini, et al. 2021. Loss-based active learning for named entity recognition. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'21)*. IEEE, 1–8.

[31] Qiao Pan, Chen Huang, and Dehua Chen. 2021. A method based on multi-standard active learning to recognize entities in electronic medical record. *Math. Biosci. Eng.* 18 (2021), 1000–1021.

[32] Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabás Póczos, and Tom Mitchell. 2019. Competence-based curriculum learning for neural machine translation. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 1162–1172.

[33] Zhicong Qiu, David J. Miller, and George Kesidis. 2016. A maximum entropy framework for semisupervised and active learning with unknown and label-scarce classes. *IEEE Trans. Neural Netw. Learn. Syst.* 28, 4 (2016), 917–933.

[34] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. 2015. Semi-supervised learning with ladder networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*. 3546–3554.

[35] Laine Samuli and Aila Timo. 2017. Temporal ensembling for semi-supervised learning. In *Proceedings of the International Conference on Learning Representations (ICLR'17)*, Vol. 4. 6.

[36] Tobias Scheffer, Christian Decomain, and Stefan Wrobel. 2001. Active hidden Markov models for information extraction. In *In Proceedings of the 4th International Conference on Advances in Intelligent Data Analysis*. 309–318..

[37] Ozan Sener and Silvio Savarese. 2018. Active learning for convolutional neural networks: A core-set approach. In *Proceedings of the 6th International Conference on Learning Representations*.

[38] Burr Settles. 2004. Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP'14)*. 107–110.

[39] Burr Settles and Mark Craven. 2008. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1070–1079.

[40] Golnar Sheikhshab, Inanc Birol, and Anoop Sarkar. 2018. In-domain context-aware token embeddings improve biomedical named entity recognition. In *Proceedings of the 9th International Workshop on Health Text Mining and Information Analysis*. 160–164.

[41] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. 2019. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5972–5981.

[42] Jianlin Su, Ahmed Murtadha, Shengfeng Pan, Jing Hou, Jun Sun, Wanwei Huang, Bo Wen, and Yunfeng Liu. 2022. Global pointer: Novel efficient span-based approach for named entity recognition. arXiv:2208.03054. Retrieved from https://arxiv.org/abs/2208.03054.

[43] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Adv. Neural Inf. Process. Syst.* 30 (2017).

[44] Xing Wu, Cheng Chen, Mingyu Zhong, Jianjia Wang, and Jun Shi. 2021. COVID-AL: The diagnosis of COVID-19 with deep active learning. *Med. Image Anal.* 68 (2021), 101913.

[45] Cao Xiao, Edward Choi, and Jimeng Sun. 2018. Opportunities and challenges in developing deep learning models using electronic health records data: A systematic review. *J. Am. Med. Inf. Assoc.* 25, 10 (2018), 1419–1428.

[46] Jinghui Yan, Yining Wang, Lu Xiang, Yu Zhou, and Chengqing Zong. 2020. A knowledge-driven generative model for multi-implication chinese medical procedure entity normalization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'20)*. 1490–1499.

[47] David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. 189–196.

[48] Tianxiang Yin, Ningzhong Liu, and Han Sun. 2021. Self-paced active learning for deep CNNs via effective loss function. *Neurocomputing* 424 (2021), 1–8.
[49] Donggeun Yoo and In So Kweon. 2019. Learning loss for active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 93–102.
[50] Jie Zhang, Dan Shen, Guodong Zhou, Jian Su, and Chew-Lim Tan. 2004. Enhancing HMM-based biomedical named entity recognition by studying special phenomena. *J. Biomed. Inf.* 37, 6 (2004), 411–422.
[51] Ningyu Zhang, Qianghuai Jia, Kangping Yin, Liang Dong, Feng Gao, and Nengwei Hua. 2020. Conceptualized representation learning for chinese biomedical text mining. arXiv:2008.10813. Retrieved from https://arxiv.org/abs/2008.10813.
[52] Shaojun Zhao. 2004. Named entity recognition in biomedical texts using an HMM model. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP'04)*. 87–90.
[53] GuoDong Zhou and Jian Su. 2002. Named entity recognition using an HMM-based chunk tagger. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. 473–480.

Received 30 November 2022; revised 9 March 2023; accepted 13 March 2023