ZoomNet for topic-oriented fragment recognition in long documents

YUKUN YAN¹, DAQI ZHENG², ZHENGDONG LU³ AND SEN SONG *
¹Biomedical Department, Tsinghua University, Beijing, 100084, China (e-mail: yanyk13@mails.tsinghua.edu.cn)
²Research Department, Deeplycurious.ai (e-mail: da@deeplycurious.ai)
³Research Department, Deeplycurious.ai (e-mail: luz@deeplycurious.ai)
*Biomedical Department, Tsinghua University, Beijing, 100084, China (e-mail: songsen@tsinghua.edu.cn)
Corresponding author: Sen Song (e-mail: songsen@tsinghua.edu.cn).

ABSTRACT This work introduces a new information extraction task called Topic-Oriented Fragment Recognition (TOFR), whose goal is to recognize information related to a specific topic in long documents from professional fields. In this paper, we introduce two TOFR datasets of judgment documents to study the problems of processing long documents. We propose a novel neural framework named Zooming Network (ZoomNet), which overcomes the challenge of combining information over long distances with limited computing resources by flexibly switching between skimming and intensive reading in processing long documents. In general, ZoomNet first establishes a hierarchical representation aligned to the text structure and then synthesizes different levels of information to assign tags via multi-scale actions. Experiments show that ZoomNet outperforms several state-of-the-art sequence labeling models on both TOFR datasets over five F1-measure.

INDEX TERMS information extraction, neural network, long documents, reinforcement learning

I. INTRODUCTION

Information extraction is a fundamental branch of natural language processing. Multiple tasks are proposed to meet the requirement of different applications. For example, part of Speech and semantic parsing is used to build structured representations for machine translation; Named entities are identified for relation analysis. From the practice of implementing NLP technology in professional domains, we observe a new requirement of information extraction that since professional documents contain a wide range of information and have a complex text structure to meet the completeness, it is necessary to first identify fragments related to a specific downstream application. Take the identifications of crime behaviors based on legal verdicts as an example. Detectives have to read through a whole document and locate crime facts before further analysis. To meet this requirement, we introduce a new information extraction task, called topic-oriented fragment recognition (TOFR), aiming to recognize related fragments given a specific topic like ‘crime fact’ and ‘clinical manifestation’.

There are two main differences between previous information extraction tasks and TOFR. First, compared to texts like news, wiki, and social media, that are used in previous works, the average length of professional documents is over 5000 tokens, and they have much more complex contextual dependency structures. Second, the target fragments in TOFR have a wider range of length variations, from one word to several sentences. To enhance larger-scale contextual information, we formulate TOFR as a document-level sequence labeling task.

Conventional approaches solve sequence labeling in an encoding-decoding manner [1] [2]. A model encodes content and context information into a distributed representation referred to as a semantic representation at each position of decision, then decodes it by assigning a tag to the corresponding token via a classification operation. Thus, the core of sequence labeling is the establishment of semantic representation. Previous methods are mainly based on recurrent neural networks or transformer-based modules. Harnessing recurrent connections and multiple gates, the former can sequentially update the semantic representation to synthesize previous and current information. However, the memory fades exponentially with increasing distances in these models. Therefore, it is difficult for a recurrent unit to model dependency across a long distance, as illustrated by several previous works [3] [4]. A transformer-based model like BERT [5] and XLNET [6] concentrates information at multiple positions with multi-head attention operations, relieving the burden of sequentially updating a semantic representation. However, they are computationally expensive,
for that the cost scales quadratically with the length of inputs in these models. As a compromise, most models have a length limitation of 512 or 1024 tokens which most professional documents are far beyond. It is very challenging for both classes of methods to model long-distance dependencies in TOFR.

We observe that when reading long documents, humans tend to skim to pick up the main ideas of sentences and paragraphs and form an outline before zooming into detailed information. In this work, we propose a novel sequence labeling model named Zooming Network (ZoomNet) to imitate this process. ZoomNet is capable of building a multi-scale representation of a document in the encoding stage, and automatically switching between skimming and intensive reading during decoding. In particular, ZoomNet casts a document into a three-level hierarchical memory aligned to the text structure. It selects a processing mode based on the state and multi-scale text information at each time step, updates the semantic representation, and processes a subsequence correspondingly. Specifically speaking, it reads a whole sentence or paragraph and outputs a tag sequence of the corresponding length under skimming mode. It processes a word when the intensive reading mode is selected. Since there is no explicit clue indicating which reading mode is better at each position, we design a step target function coupled with a reward corresponding to the ratio of multi-level actions to train our model. In general, the objective function drives the model to achieve high labeling accuracy with as little intensive reading as possible.

Experimental results show that our model outperforms existing state-of-the-art sequence labeling models on both TOFR datasets. Additionally, we find that the processing mode of ZoomNet is similar to a human reader. That is to say, ZoomNet can flexibly switch between intensive reading and skimming to achieve good performance.

Our main contribution can be summarized as follows:

- We propose a new information extraction task called Topic-Oriented Fragment Recognition to address the recognition of topic-relevant fragments in long professional documents and study large-scale contextual information modeling, and we introduce two datasets based on judgment documents for TOFR.
- We propose an encoding method to establish a multi-scale representation of a document. This storage mode

FIGURE 1. Samples of TOFR. We omit some parts of the original text for simplification and provide English Translation below. The target fragments are marked red in the texts. The goal of a TOFR task is to extract fragments, which are related to a specific topic corresponding to downstream applications, from professional documents.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
relieves the conflict between local information and extensive contextual information.

- We propose a novel neural network model capable of flexibly switching between intensive reading and skimming to process a document, like human readers.
- Our model outperforms other sequence labeling models, including BiLSTM-CRF, BERT, XLNET, RoBERTa, and ELECTRA on both TOFR datasets with a large margin.

II. RELATED WORK

Sequence Labeling In the NLP domain, sequence labeling is a pattern recognition task that requires a system to assign a categorical label to each token or other language unit. Examples of classical tasks include part-of-speech (POS) tagging [7], named entity recognition (NER) [8] [9], text chunking [10], etc. which is the foundation of many downstream applications like entity linking, coreference resolution, etc. However, in most tasks, the samples are processed in the form of sentences which limits the contextual information to a sentence. On the other hand, the topic-oriented fragment recognition proposed in this paper focuses on evaluating models in modeling document-level context, which is required in many real-world applications.

Hierarchical Encoding Compared to an ungraded encoding strategy, building a hierarchical representation brings several advantages: (1) The model can separately store multiple levels of contextual information to alleviate their conflict with each other. (2) A deeper structure can support more flexible update methods to avoid losing information by shortening the dependency distance. Towards implementing hierarchical encoding, researchers have proposed some pioneering attempts such as Hierarchical RNN [11], NARX [12], and clockwork RNN (CW-RNN) [13] introduce different fixed update frequencies with multiple RNN layers. Gated Feedback Recurrent Neural Network [14], Hierarchical Multiscale Recurrent Neural Networks [15], Parsing-Reading-Predict Networks (PRPN) [16] introduce a variety of gating units between the lower-level layer and upper-level layer. In Focused Hierarchical RNN [17], the connections between the lower-level layer and upper-level layer are controlled by a hard gate regulated on a Bernoulli distribution whose parameter is determined by a special function applied on context and question. However, these models mainly focus on syntactic features within a short context and ignore large-scale document structures and global information.

The model that inspired us the most is Hierarchical Attention Networks (HAN) [18], a text classification model proposed by Yang et al. It uses a bottom-up strategy to encode a text. Specifically, HAN leverages two-level product attention to generate the representation of sentences and documents. In a similar way, our model establishes a hierarchical representation aligned to word, sentence, and paragraph levels respectively. However, we use max-pooling instead of product attention in building an upper-level representation for two reasons: (1) Applying attention to long documents brings a huge computation burden. (2) Unlike in the classification task, it is difficult to collect all useful contextual information with a single query in the TOFR task.

Another model proposed by Seunghyun et al [19] also builds a two-level representation in a bottom-up way. In addition, it leverages sentence-level supervision to help train the model in a sequence labeling task. It illustrates that by learning to perform tasks jointly on multiple levels, the model achieves substantial improvements for sequence labeling. However, the sentence-level supervisions it used are labeled by hand. Such labels are expensive to add in most situations, and not commonly available in sequence labeling tasks. In contrast, the sentence-level and paragraph-level labels used to train ZoomNet are automatically generated from those at the word level. In addition, our model leverages multi-level actions in decoding progress.

Attention based Models Many recent models like variants of BERT [5] and XLNET [6] based on stacked transformer encoders leverage self-attention to collect information from different positions. Following a pre-training strategy, they achieve state-of-the-art performances on several sequence labeling tasks. Nevertheless, they require a large number of computing resources when processing long documents and a large number of samples to train. The most successful model on sequence labeling tasks is SpanBERT [20]. It introduces a novel pre-training task called span-boundary objective (SBO), which makes it suitable for encoding span-level information. However, the spans used to train SpanBERT are limited to phrase-level, which are much shorter than the target fragments in the proposed TOFR tasks. In addition, same as other models leverages different pre-training strategies like ALBERT [21], ELECTRA [22], and RoBERTa [23], it shares the same model structure of original BERT.

Sequence Labeling on Long-Distance Documents Due to the lack of appropriate datasets, there are few works on sequence long-document sequence labeling. Jörke et al. proposed a model [24] that learns to attend to multiple mentions of the same word type in generating a representation for each token in context to implement named entity recognition (NER) on long documents. However, this paradigm is not suitable for TOFR, since the target fragments can be much longer, and it is hard to determine the corresponding context.

Processing Modes There are also some related works that learn to skim the text to accelerate the prediction process [Yu et al [25], Liu et al. [26]]. For example, Liu et al. [26] propose to learn to skip all the future sentences after the model has made a decision. However, these models are mainly targeting text classification instead of sequence labeling, and they cannot be generalized to long documents.

Decoding Actions To our best knowledge, our model is the first one that employs the document structure in decoding. Leveraging multi-level decoding actions, the proposed model can choose contextual information more flexibly to update the decoding state in a sequence labeling task.

III. METHOD
A. OVERVIEW

In this section, we start with a brief overview of the encoding and decoding process of ZoomNet and then proceed to structure the following sections with more details about the submodules of our model namely Hierarchical Encoder and Action Decoder as shown in Figure 2.

**Encoding**

Given a document $D$ with $N$ words, $M$ sentences, and $L$ paragraphs (The boundaries of sentences and paragraphs are identified by punctuations and line breaks), ZoomNet casts it into a hierarchical representation $H$ consisting of three components:

$$H = \{W, S, P\}$$

$$W = \{w_1, w_2, ..., w_N\}$$

$$S = \{s_1, s_2, ..., s_M\}$$

$$P = \{p_1, p_2, ..., p_L\}$$

After encoding, each word, sentence, and paragraph has a corresponding vector representation in $H$.

**Decoding**

We introduce nine decoding actions at three scales in ZoomNet, as shown in Figure 3. Each one corresponds to a tag type according to the BIO schema and a labeling window whose size is equal to the current word, sentence, or paragraph. With sentence/paragraph-level actions, ZoomNet can label more than one word at a single time step. In addition, large-scale actions enable the model to update its state without attending to all detailed information.

Specifically speaking, at each time step, ZoomNet takes as input the concatenation of a word, a sentence, and a paragraph, indicated by the location vector $l_t = [i, j, k]$. It maintains a decoding state $h_t$ with an LSTM layer and predicts an action $a_t$ at each time step. The action is then used to update the location vector.

$$l_t = [i, j, k]$$

$$x_t = [w_i, s_j, p_k, a_{t-1}]$$

$$h_t = f(x_t, h_{t-1})$$

$$y_t = FFNN(x_t)$$
The representation of a paragraph $p_k$ is generated in a similar way to the representation of the sentence by a third bidirectional LSTM layer with the whole document as the context:

$$z^p_k = \max(s_1, s_2, \ldots, s_{L_k})$$  \hspace{1cm} (18)

$$\overrightarrow{h^k_p} = \text{LSTM}^p(\overrightarrow{h^p_{k-1}}, z^p_k)$$  \hspace{1cm} (19)

$$\overrightarrow{h^k_p} = \text{LSTM}^p(\overrightarrow{h^p_{k+1}}, z^p_k)$$  \hspace{1cm} (20)

$$p_k = [\overrightarrow{h^k_p}, \overrightarrow{h^k_p}]$$  \hspace{1cm} (21)

Finally, we combine all the representations at the three levels to form the hierarchical representation as shown in equation (1).

**Action Decoder** The action decoder consists of an LSTM layer and a feedforward layer as shown in the right part of Figure 2. It decodes the hierarchical representation by sequentially generating decoding actions as shown in Figure 3.

Similar to previous works like [27] [28] [29], the decoder of our model uses an LSTM layer to maintain a decoding state. However, different from these models, instead of taking encoded vectors one by one or implementing dot-product attention on all positions [30] [31] [32], it takes as input the information at multiple levels indexed by a vector $l_t \in Z^3$ consisted of three indexing numbers. As feedback, the action at the previous time step is cast into a one-hot vector $v^a_{t-1}$ and also feed into the LSTM unit to update the decoding state.

Following classic sequence to sequence models, the decoding state is transformed into $y_t \in R^q$ for the final softmax over the nine decoding actions. During the training stage, the action is sampled according to the probability calculated with equation (29) to implement reinforcement learning, and it is determined by equation (30) during evaluation.

$$y_t = W \cdot h_t + b$$  \hspace{1cm} (22)

$$p_t(a_i) = \text{softmax}(y_{t,i})$$  \hspace{1cm} (23)

$$a_t = \arg \max_i (p_t(a_i))$$  \hspace{1cm} (24)

Each decoding action has a label type and a labeling window as illustrated in Figure 3. The beginning boundary of its labeling window is the indexing number of the current word and the end index corresponds to the end of a word, a sentence, or a paragraph depending on the type of action. The decoding action also updates the position vector $l_t$ by making it point to the word next to its labeling window.
Action Combinations

1. This is the first sentence. This is the second sentence. This is a new paragraph.
   \( B_u, l, O_u, O_u, O_u, O_u, O_u, O_u, O_u \)
2. This is the first sentence. This is the second sentence. This is a new paragraph.
   \( B_u, I_u, I_u, I_u, I_u, O_u, O_u, O_u, O_u, O_u, O_p \)
3. This is the first sentence. This is the second sentence. This is a new paragraph.
   \( B_u, O_u, O_u \)

Predicted Label Sequence
\( \{B, I, I, I, I, I, O, O, O, O, O, O, O, O, O, O, O, O, O, O\} \)

**FIGURE 4.** Different action combinations with the same label sequence. In ZoomNet, the same label sequence can be generated with different decoding action combinations. A paragraph can be processed with only a few actions like the third example or with multiple word-level actions like the second example.

### C. TRAINING

In general, a sequence labeling model is trained by maximizing the probability of the correct sequence of labels [33] [34] [35]. Nevertheless, the correct decoding action sequence is not unique in ZoomNet as different action combinations can generate the same label sequence as shown in Figure 4. There is no explicit supervision of what information should be processed word by word and what should be skimmed. Therefore, we modified the computation of cross-entropy to maximize the probability of the correct actions set at each time step.

Intuitively, skimming can help a reader avoid the interferences of target-irrelevant information in processing large-scale decoding actions as long as they produce correct tag sequences. Specifically speaking, at the end of processing each sample, the proposed model gains a reward to balance the accuracy and the exploration of new action combinations.

Therefore, we encourage our model to use more large-scale decoding actions as long as they produce correct tag sequences. Specifically speaking, at the end of processing each sample, the proposed model gains a reward based on the ratio of word-level actions and all the actions used. We update the parameters using a policy-gradient strategy.

Since there may be more than one correct action at each time step, we use a target function modified from the cross-entropy to maximize the sum probability of all correct actions:

\[
J(\theta) = E(\theta) \left[ \sum_{t=1}^{T} r_t \right] \quad (26)
\]

The policy gradient method solves the maximization problem using gradient ascent. That is, we keep stepping through the parameters using the following update rule:

\[
\theta_{\text{new}} = \theta_{\text{old}} + \alpha \nabla J(\theta_{\text{old}}) \quad (27)
\]

Where \( \theta \) represents the parameters of a policy, and \( \alpha \) is the learning rate. In this work, the policy \( \pi(a|s, \theta) \) is computed in equation (23).

Intuitively, a model capable of imitating a human reader should use intensive reading only when processing important information to save memory. Thus, the reward should be negatively associated with the ratio of word-level actions in each episode. Here we give the model a reward \( r^e \) after processing each sample as follow:

\[
r^e = -\frac{N_w}{N_w + N_s + N_p} \quad (28)
\]

Where \( N_w, N_s, \) and \( N_p \) represent the number of corresponding levels of actions.

We limit the exploration of actions combinations to those that produce correct label sequences. That is, the proposed model randomly executes a decoding action from the correct action set \( A_t^* \) at each time step aligned to the normalized probabilities in equation (29).

\[
p_t^c(a_i) = \frac{p_t(a_i)}{\sum_{a_j \in A_t^*} p_t(a_j)} \quad (29)
\]

As aforementioned, we apply supervised learning at each time step and update the parameters following a policy-gradient method after the processing of each sample. Nevertheless, it is impossible to simultaneously minimize the loss function in equation (25) and maximize the reward in equation (28) that some target fragments can only be recognized by word-level actions. Thus, we introduce a hyper-parameter \( \lambda \) to balance the accuracy and the exploration of new action combinations.

\[
J(\theta) = J(\theta) \cdot r^e \sum_{t=1}^{T} \log p_t(a_i) \quad (30)
\]

Where \( J(\theta) \) is the target function we maximize in practice. The policy-gradient method is implemented following the update rule in equation (27).

### IV. EXPERIMENTS

#### A. DATASETS

In this paper, we introduce two TOFR datasets both are labeled by five members of an annotation group in Deeply-curious.AI trained by two expert lawyers with the necessary knowledge to recognize the target fragments. Each document is annotated by two individuals separately. The labels with
divergences are passed to a third annotator to give a final decision.

| Data Set | CDR | DDR |
|----------|-----|-----|
| sample size | 5000 | 2335 |
| min length | 788 | 1442 |
| max length | 14151 | 27964 |
| average length | 1956.61 | 5640.33 |
| average target fragment number | 1.30 | 1.44 |
| average target fragment length | 100.39 | 24.61 |

TABLE 1. Statistics of the data sets.

The first dataset, criminal description recognition (CDR), is based on judgment documents of theft cases. As illustrated in the upper part in Figure 1, it requests a model to identify fragments that describe the criminal behaviors, including date, address, time, victims, tools, etc. CDR consists of 5000 samples, with an average length of 1957 tokens.

The second dataset, dispute description recognition (DDR), is based on judgment documents concerning intellectual properties. As shown in Figure 1, the target fragments are the disputes in the court, which can be either detailed descriptions or conclusive sentences. There are 2335 samples in DDR, with an average length of 5640 tokens. More detailed statistics of the datasets are shown in TABLE 1.

We evaluate the performance of baseline models and our model on both two TOFR datasets with 80% of the data for training, 10% for validation, and the remaining 10% for testing.

**Schema** We formulate TOFR as a sequence labeling task: given a sequence \( X = \{x_1, x_2, ..., x_N\} \) with \( N \) tokens, it requires a model to generate corresponding labels \( Y = \{y_1, y_2, ..., y_N\} \) with equal length to indicate the target fragments. There are three types of tags: (1) B-tag: the beginning token of a topic-related segment. (2) I-tag: a token inside a topic-related segment. (3) O-tag: a token belongs to no topic-related segment. An example is shown in Figure 5 to illustrate the BIO schema used in TOFR tasks.

**Evaluation Metrics** We evaluate the performance on TOFR in terms of precision, recall, and F1-score, where: (1) Precision is the fraction of correct fragments recognized among the retrieved ones; (2) Recall is the fraction of correct fragments that are identified. (3) F1-score is the harmonic mean of precision and recall. A target fragment is correctly identified only if both starting and end indices exactly match the ground truth labels.

B. BASELINES

We compare our method with several state-of-the-art sequence labeling approaches, including RNN-based and transformer-based models:

- **BiLSTM-CRF** [34]: This is the first model to apply a bidirectional LSTM CRF to NLP benchmark sequence tagging data sets. It can efficiently use both past and future input features thanks to a bidirectional LSTM component. For a fair comparison, we set the number of bidirectional LSTM layers to three, the same as the proposed model.
- **BERT** [5]: BERT stands for Bidirectional Encoder Representations from Transformers. It is designed to pre-train deep bidirectional representations from unlabeled texts by jointly conditioning on both left and right context in all layers. Normally, it can be fine-tuned with just one additional output layer to meet specific tasks.
- **XLNET** [6]: XLNET is a generalized autoregressive pre-training method based on Transformer-XL that (1) enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order and (2) overcomes the limitations of BERT thanks to its autoregressive formulation. In this work, we fine-tune the model based on [36].
- **RoBERTa** [23]: RoBERTa has the same structure as BERT model, and it is trained with dynamic masking, FULL-SENTENCES without NSP loss, large mini-batches, and a larger byte-level BPE. It achieves better performance on multiple NLP tasks compared to BERT. We use the pre-trained model introduced in [36].
- **ELECTRA** [22]: ELECTRA is pre-trained with the task called replaced token detection, which is more efficient than masked language modeling (MLM). The contextual representations learned substantially outperform the ones learned by BERT given the same model size, data, and compute. The model is initialized with...
pre-trained parameters in the work [36].

We did not compare the proposed model to other sequence labeling models like [24] for that though they are designed to enhance long-distance contextual information, they are limited to the recognition of phrase-level targets like named entities. In addition, we propose a variant of the proposed model by generating the word-level representation with a pre-trained BERT model.

C. PARAMETERS SETTING

| Hyper-Parameter                        | Value |
|----------------------------------------|-------|
| word embedding size                    | 64    |
| LSTM layer dimension                   | 256   |
| Model size (for Transformer-based models) | 768   |
| FFNN dimension                         | 256   |
| Word-level BiLSTM dimension            | 128   |
| Sentence-level BiLSTM dimension        | 256   |
| Paragraph-level BiLSTM dimension       | 512   |
| \(\lambda\)                            | 1     |
| Optimizer                              | Adam  |
| Fine-tuning learning rate              | 1e-6  |
| Other learning rate                    | 1e-3  |

TABLE 2. Major hyper-parameters for Implemented models

We built the vocabulary by retaining words that appear 20 times and replacing the other words with a special [unknown] token. The parameters in the embedding layer are randomly initialized and kept trainable during training. The dimension of embedding vectors is set to be 64, and the dimensions of the three-level LSTM layers in the encoder are 128, 256, 512. The dimension of LSTM in the decoder and the hidden layer in the classifier are set to be 128. The weight parameter is tuned to be 0.5 on the validation set. For training, we use a mini-batch size of 48 and an Adam optimizer with an initial learning rate of 1e-5. The detailed hyper-parameter setting can be found in Table 2. The source code of our work is available here(Pytorch 1.14).

D. OVERALL RESULTS

We report the performance, including precision, recall, F1-score as illustrated in section III, and the ratio of word-level actions in Table 3. ZoomNet outperforms baseline models with a significant advantage on both tasks. In addition, the BERT-based model achieves better performance than the original one.

BiLSTM-CRF did not achieve a competitive performance with other models on both TOFR tasks, indicating that LSTM is incompetent in processing long texts, which is consistent with several previous studies [4]. BERT, XLNET, and the other two transformer-based models achieve similar performances. Compared with BiLSTM-CRF, they are more powerful in modeling complex dependencies. However, with a limited input window size set by the pre-trained configurations, they can not take as input a whole long document. Since the document is divided, it is likely to lose important contextual information, and even the target fragments can break into different segments, which is usually found in error cases. In contrast, with the same number of LSTM layers as used in BiLSTM-CRF, the proposed model enhances long contextual information via a hierarchical encoding strategy and is free of text length limitation. The original version of ZoomNet and BERT-based one achieve similar performance on both tasks. Furthermore, bidirectional Transformer does not show obvious superiority compared to BiLSTM coupled with a random initialized word embedding matrix under our hierarchical encoding strategy.

E. READING MODES ANALYSIS

In this work, the proposed model is driven to mimic the reading pattern of human readers, capable of selectively concentrating or skimming different parts of the text. We use the mean ratio of word-level actions \(R_w^a\), as shown in Table 3 and Table 4 on the test sets to estimate how much part of the text is considered detail information.

\[
R_w^a = \frac{N_w}{N_w + N_s + N_p}
\]

(31)

Where \(N_w\), \(N_s\), and \(N_p\) represent the number of corresponding levels of actions. We show an instance of the changes of F1-score together with \(R_w^a\) on the test set of the proposed model in a training trail on task I in Figure 6. As illustrated, \(R_w^a\) keeps fluctuating in the beginning for that the value of the supervised target function remains large. And as the performance becomes stable, the model tends to decrease the usage of word-level actions. That is, the proposed model learns to use less detailed information to achieve comparable accuracy. This is similar to the learning process of a human reader, who gradually intimacies with professional knowledge.

FIGURE 6. \(R_w^a\) and F1-score during training. \(R_w^a\) keeps fluctuating in the beginning for that the absolute value of supervised target function remains large. And as the performance becomes stable, the model tends to decrease the usage of word-level actions.

Furthermore, a process instance is shown in Figure 7 to better illustrate how the proposed model switch between...
TABLE 3. Evaluation Results ($R^w$ stands for the ratio of word-level actions). ZoomNet outperforms the baseline models with significant margins.

| Models          | Precision | Recall | macro F1 | $R^w$ (%) | Precision | Recall | macro F1 | $R^w$ (%) |
|-----------------|-----------|--------|----------|----------|-----------|--------|----------|----------|
| BiLSTM-CRF [34] | 78.62     | 85.26  | 81.81    |          | 78.31     | 86.83  | 82.36    |          |
| BERT [5]        | 87.23     | 90.72  | 88.94    |          | 86.92     | 92.55  | 89.64    |          |
| XLNET [6]       | 88.15     | 90.97  | 89.53    |          | 88.23     | 91.93  | 90.04    |          |
| RoBERTa [23]    | 88.24     | 96.96  | 92.40    |          | 89.34     | 94.33  | 91.76    |          |
| ELECTRA [22]    | 88.36     | 95.29  | 91.69    |          | 88.66     | 92.75  | 90.65    |          |
| ZoomNet         | 95.34     | 95.64  | 95.49    | 16.13    | 94.74     | 95.06  | 94.91    | 15.22    |
| ZoomNet(BERT-based) | 95.56   | 95.73  | 95.64    | 16.13    | 94.88     | 95.10  | 94.98    | 15.38    |

TABLE 4. Ablation Results. Ablation study on action space and the balance between supervised learning and reinforcement learning.

| Variable | Line Action Space | Precision | Recall | macro F1 | $R^w$ (%) |
|----------|-------------------|-----------|--------|----------|----------|
| Action Space | 1 word         | 91.02     | 88.39  | 89.68    | 100      |
|           | 2 +sentence      | 95.12     | 93.54  | 94.02    | 2.97     |
|           | 3 +paragraph     | 95.34     | 95.64  | 95.49    | 16.13    |
| $\lambda$ | 4 0              | 97.21     | 85.13  | 90.77    | 65.89    |
|           | 5 0.2            | 95.32     | 90.44  | 92.82    | 41.32    |
|           | 6 1              | 95.34     | 95.64  | 95.49    | 16.13    |
|           | 7 5              | 93.10     | 93.11  | 93.11    | 2.53     |

F. ABLATION STUDY

To evaluate the contribution of hierarchical encoding strategy, multi-scale action space, and the hybrid of supervised and reinforcement learning, we test the performances of ablated models on the CDR dataset. The results are listed in Table 4. Line 1 in Table 4 shows that our model achieves 90.77 F1-scores without sentence-level and paragraph-level actions. It yields a 4.5 percent F1 score improvement if we expand the action space to the sentence level and another 1 percent F1 score improvement if we add paragraph-level actions. Line 4-7 of Table 4 illustrates that $\lambda$ affects the ratio of intensive reading and skimming significantly. From the observation of error cases, we find that with more intensive reading, the fragments recognized have more precise boundaries. However, it is more likely to fail to capture important large-scale contextual information, which leads to a low recall.

V. CONCLUSION

In this paper, we focus on the extraction requirement of specific information on long professional documents. To study this problem, we propose a new information extraction task, Topic-Oriented Fragment Recognition (TOFR) that aims to recognize fragments related to specific topics from professional documents and two corresponding datasets. Inspired by the reading pattern of human readers, we propose a novel model, named Zooming Network (ZoomNet), which is capable of generating a hierarchical representation and leveraging multi-scale decoding actions to complete sequence labeling. Experiments show that our model outperforms several state-of-the-art sequence labeling models, including BiLSTM-CRF, BERT, XLNET, RoBERTa, and ELECTRA on both datasets with a big margin. For future work, we will apply the proposed model to other natural language processing tasks, like machine reading comprehension, text classification, on skimming and intensive reading. ZoomNet quickly skims the first several sections, including basic information on the criminal, investigation procedure of police. Once reaching the part that contains detailed clues or is near target fragments, it starts to process word by word and tends to skim after all target fragments are recognized.
long documents.

REFERENCES

[1] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in Advances in neural information processing systems, 2014, pp. 3104–3112.

[2] R. Dey and F. M. Salem, “Gate-variants of gated recurrent unit (gru) neural networks,” in 2017 IEEE 60th international microwave symposium on circuits and systems (MWS CS). IEEE, 2017, pp. 1597–1600.

[3] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” IEEE transactions on neural networks, vol. 5, no. 2, pp. 157–166, 1994.

[4] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in NAACL-HLT (1), 2019.

[6] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, “Xlnet: Generalized autoencoder pretraining for language understanding,” Advances in neural information processing systems, vol. 32, 2019.

[7] V. Giat, L. Bierkandt, S. Druskat, and C. Ryzinski, “Enriching timebank: Towards a more precise annotation of temporal relations in a text,” in Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16), 2016, pp. 3844–3850.

[8] N. Peng and M. Dredze, “Named entity recognition for chinese social media with jointly trained embeddings,” in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 548–554.

[9] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, and H. Poon, “Domain-specific language model pretraining for biomedical natural language processing,” arXiv e-prints, pp. arXiv–2007, 2020.

[10] E. F. Sang and S. Buchholz, “Introduction to the conll-2000 shared task: Chunking,” arXiv preprint cs/0009008, 2000.

[11] S. El Hhi and Y. Bengio, “Hierarchical recurrent neural networks for long-term dependencies,” in Advances in neural information processing systems, 1996, pp. 493–499.

[12] T. Lin, B. G. Horne, P. Tino, and C. L. Giles, “Learning long-term dependencies in naru recurrent neural networks,” IEEE Transactions on Neural Networks, vol. 7, no. 6, pp. 1329–1338, 1996.

[13] J. Koutnik, K. Greff, F. Gomez, and J. Schmidhuber, “A clockwork rnn,” in International Conference on Machine Learning. PMLR, 2014, pp. 1863–1871.

[14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Gated feedback recurrent neural networks,” in International conference on machine learning, PMLR, 2015, pp. 2067–2075.

[15] J. Chung, S. Ahn, and Y. Bengio, “Hierarchical multiscale recurrent neural networks,” in 5th International Conference on Learning Representations, ICLR 2017, 2019.

[16] Y. Shen, Z. Lin, C.-W. Huang, and A. Courville, “Neural language modeling by jointly learning syntax and lexicon,” arXiv preprint arXiv:1711.02013, 2017.

[17] N. R. Ke, K. Zoha, A. Sordoni, Z. Lin, A. Trischler, Y. Bengio, J. Pineau, L. Charlin, and C. Pal, “Focused hierarchical rnns for conditional sequence processing,” in International Conference on Machine Learning. PMLR, 2018, pp. 2554–2563.

[18] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, “Hierarchical attention networks for document classification,” in Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, 2016, pp. 1480–1489.

[19] M. Rei and A. Søgaard, “Jointly learning to label sentences and tokens,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, 2019, pp. 6916–6923.

[20] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, “Spanbert: Improving pre-training by representing and predicting spans,” Transactions of the Association for Computational Linguistics, vol. 8, pp. 64–77, 2020.

[21] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricet, “Albert: A lite bert for self-supervised learning of language representations,” arXiv e-prints, pp. arXiv–1909, 2019.

[22] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, “Electra: Pre-training text encoders as discriminators rather than generators,” arXiv preprint arXiv:2003.10555, 2020.

[23] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” arXiv preprint arXiv:1907.11692, 2019.

[24] M. Jörke, J. Gillick, M. Sims, and D. Bamman, “Attending to Long-Distance Document Context for Sequence Labeling,” in Findings of the Association for Computational Linguistics: EMNLP 2020. Online: Association for Computational Linguistics, Nov. 2020, pp. 3692–3704. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.330

[25] A. W. Yu, H. Lee, and Q. V. Le, “Learning to skim text,” arXiv preprint arXiv:1704.06877, 2017.

[26] X. Liu, L. Mou, H. Cui, Z. Lu, and S. Song, “Jumper: Learning when to make classification decision in reading,” in IJCAI, 2018.

[27] Y. Zhang and J. Yang, “Chinese ner using lattice lstm,” in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2018, pp. 1554–1564.

[28] N. Limsopatham and N. Collier, “Bidirectional lstm for named entity recognition in twitter messages,” 2016.

[29] B. Settles and M. Craven, “An analysis of active learning strategies for sequence labeling tasks,” in Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, 2008, pp. 1070–1079.

[30] X. Ma and E. Hovy, “End-to-end sequence labeling via bi-directional lstm-cnns-crf,” in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016, pp. 1064–1074.

[31] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” arXiv preprint arXiv:1409.0473, 2014.

[32] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder–decoder approaches,” in Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, 2014, pp. 103–111.

[33] A. Ramponi, R. van der Goot, R. Lombardo, and B. Plank, “Biomedical event extraction as sequence labeling,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 5357–5367.

[34] Z. Huang, W. Xu, and K. Yu, “Bidirectional lstm-crf models for sequence tagging,” arXiv preprint arXiv:1508.01991, 2015.

[35] F. Rodrigues, F. Pereira, and B. Ribeiro, “Sequence labeling with multiple annotators,” Machine learning, vol. 95, no. 2, pp. 165–181, 2014.

[36] Y. Cui, W. Che, T. Liu, B. Qin, S. Wang, and G. Hu, “Revisiting pre-trained models for Chinese natural language processing,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings. Online: Association for Computational Linguistics, Nov. 2020, pp. 657–668. [Online]. Available: https://www.aclweb.org/anthology/2020.findings-emnlp.58

YUKUN YAN is currently a Ph.D. student at Tianjin University. He received B5 degree in the School of Electronic Information Engineering, Tianjin University. He mainly focuses on deep learning and applications in natural language processing (NLP) tasks, such as document classification and sequence labeling.
DAQI ZHENG  Head of Planning Department, DeeplyCurious.AI. Graduate from Institute of Computing Technology, Chinese Academy of Sciences. Research interests focus on natural language processing and machine learning.

ZHENGDONG LU  is the founder and CTO of DeeplyCurious.AI (a startup in Beijing, China). Lu got his Ph.D. degree from Oregon Health & Science University in 2008. Dr. Lu then worked as a postdoctoral researcher at the University of Texas at Austin for several years. Before creating DeeplyCurious.AI, he worked as an associate researcher in MSRA and senior researcher at Noah’s Ark Lab, Huawei. Dr. Lu has published more than 60 top conference and journal articles and has served as a reviewer for several international conferences (NIPS, ICML, and IEEE transaction on PAMI). His research interests include machine learning, deep learning, natural language understanding, and data mining.

SEN SONG  is an associate professor at Tsinghua University. He got his Ph.D. degree from Brandeis University in 2002 and worked as a postdoctoral researcher at Cold Spring Harbor Laboratory and MIT. He joined Tsinghua University 2010. His research interests include brain-inspired computing, computational neuroscience, and deep learning.

* * *