Pattern Recognition Letters 145 (2021) 157–164

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Short communication

Self-attention-based conditional random fields latent variables model for sequence labeling

Yinan Shao[a], Jerry Chun-Wei Lin[b,*], Gautam Srivastava[c,d], Alireza Jolfaei[e], Dongdong Guo[a], Yi Hu[a]

[a] Alibaba Inc., Hangzhou, Zhejiang, China
[b] Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway
[c] Department of Mathematics and Computer Science, Brandon University, Brandon, Canada
[d] Research Centre of Interneural Computing, China Medical University, Taichung, Taiwan
[e] Department of Computing, Macquarie University, Sydney, Australia

ARTICLE INFO

Article history:
Received 2 June 2020
Revised 3 February 2021
Accepted 12 February 2021
Available online 17 February 2021

MSC:
68T50
68U15
68U20
68T30

Keywords:
Latent CRF
Sequence labeling
Encoding schema
Natural language processing
VQA
Big data

ABSTRACT

To process data like text and speech, Natural Language Processing (NLP) is a valuable tool. As on of NLP's upstream tasks, sequence labeling is a vital part of NLP through techniques like text classification, machine translation, and sentiment analysis. In this paper, our focus is on sequence labeling where we assign semantic labels within input sequences. We present two novel frameworks, namely SA-CRFLV-I and SA-CRFLV-II, that use latent variables within random fields. These frameworks make use of an encoding schema in the form of a latent variable to be able to capture the latent structure in the observed data. SA-CRFLV-I shows the best performance at the sentence level whereas SA-CRFLV-II works best at the word level. In our in-depth experimental results, we compare our frameworks with 4 well-known sequence prediction methodologies which include NER, reference parsing, chunking as well as POS tagging. The proposed frameworks are shown to have better performance in terms of many well-known metrics.

1. Introduction

The first step in text processing (i.e., Natural Language Processing, NLP) is most often sequence labeling (SL). As defined, sequence labeling is the task by which semantic labels are identified and assigned to each unit within input sequences [1]. SL is also widely applied in visual question answering (VQA) [2] that considers the visual context vector and semantic information of the sentences by predicting the labels of sequence. Commonly seen labels include chunk labels, named-entity labels, part-of-speed labels. Such labels can help the model understand the semantic structure of the questions, and generate a smooth and coherent answer in VQA. Through reliance on downstream tasks, sequence labeling methods have become very popular recently in both academic research and industry. Most existing models combine CNN-based architectures and conditional random fields (CRF) with latent variables to analyze the image content to obtain the solutions in VQA. Furthermore, SL can help give components context to better understand its meaning and has been also considered a major research issue in VQA.

Historically, SL is usually achieved using entity recognition through:

1. Extractions of entity names (person names, companies, etc)
2. Chunking to find parts of sentences (verbs, nouns, adjectives)
3. Reference parsing that can extract information (author, journal, title)

Conditional random fields (CRF), as well as maximum entropy model (MEM), are types of conventional sequence labeling models that study conditional probability over input sequences. In contrast, segmentation models like semi-Markov random fields (semi-CRF) are used to represent the span of text for input sequences. Ratinov et al. showed that most encoding schemas are strongly af-

* Corresponding author.
E-mail addresses: jerrylin@ieee.org (J.C.-W. Lin), srivastavag@brandonu.ca (G. Srivastava), alireza.jolfaei@mq.edu.au (A. Jolfaei).

https://doi.org/10.1016/j.patrec.2021.02.008

BIO encoding	Geroge	Bush	votes	for	Jack
	B-PER	I-PER	O	O	B-PER
BILOU encoding	Geroge	Bush	votes	for	Jack
	B-PER	I-PER	O	O	U-PER

Fig. 1. The BIO and BILOU encoding schemas.

fected by model performance [3]. Different encoding scheme are presented in Fig. 1. Here, one can see how the different parts of BILOU are chosen.

The performance of conventional sequence labeling models largely depends on the encoding schema and feature engineering [4]. A difference in encoding schemas and feature engineering will in turn lead to different performance on different sequence labeling tasks as well as datasets. It is time-exhausting to find the best settings for every sequence labeling task and datasets. In this work, we present 2 end-to-end self-attention-based CRFs with latent variables(respectively named SA-CRFLV-1 and SA-CRFLV-2), which can automatically extract features and choose the best encoding schema for a set input on different natural language tasks and datasets. The proposed model utilizes self-attention based neural networks to extract neural features for the input sentence [5,6]. The extracted neural features are combined with hand-craft features and computed in CRFs. The CRFs take encoding schema as a latent variable and tuning during training. The first presented model known as SA-CRFLV-I can label the input using 2 encoding schemes at the same time while still optimizing parameters. In the second designed SA-CRFLV-II model, it can choose encoding schema on the word level as opposed to the sentence level which hybrids across 2 encoding schemas. Our contributions are summarized as follows:

- The designed SA-CRFLV-I and SA-CRFLV-II are the end-to-end frameworks for sequence labeling.
- Encoding schema is administered in the form of latent variables to be able to capture structures of the hidden variables as well as the observed data.
- Self-attention based models are utilized to automatically extract features for the state-of-the-art CRFLV-I and CRFLV-II in different scenarios.
- Our experimental results showed that our schemas hold strongly performance against BIO or BILOU encoding schema.

2. Literature review

We summarize what are known as “traditional” models to include (HMM - Hidden Markov Model) [7–9], conditional random field model (CRF) [10], semi-Markov random field model (semi-CRF) [11], as well as max-entropy model (MEM) [12]. All of the above-mentioned models are linear that are known to capture the correlations between labels that neighbour each other to create the best chain of labels.

CRF models [10] are the most commonly used models for sequence labeling. These well-known models exemplify a well-used class of statistical methods for modeling which have been often shown to apply for solving sequence prediction problems. In the models, there are several advantages over using just run of the mill HMM as well as stochastic grammars which include the ability for the relaxation of strong independent assumptions that are made on these models. Tseng [13] defined a Chinese-word-segmentation (CWS) system that is based solely on Conditional random field models. Zhao [14] considered the Chinese-word-segmentation problem and simplified it to a character tagging problem under strict use of the conditional random field. The authors combined feature-template with tag-set selection to enhance model performance. Cuong et al. [15] proposed efficient inference algorithms to handle high-order dependencies between labels or segments. They demonstrate that exploiting high-order dependencies can effectively enhance model performance.

Muis et al. [16] designed a weak semi-Markov CRF for use in noun-phrase based chunking. In classic semi-CRF, the model is known to intuitively decide both lengths as well as the type of next segments at the same time. However, in weak semi-CRF, the model attempts to give a weaker variant that can make these 2 decisions separate through restriction of every node which connects to other nodes only or nodes with the same label in the next segment, or every node within the next word. The weak semi-CRF model was shown to yield similar performance to classical semi-CRFs, however, runtimes were significantly better. Lin et al. [17] propose LVCRF-I and LVCRF-II which utilize encoding schema as latent variables to capture the latent structure of the hidden variables and the observed data. The performance of these two models largely depends on hand-craft features which result in poor robustness over different sequence labeling tasks and datasets.

When focusing on deep learning-based models, there have been advantages shown when considering sequence labeling tasks [18]. Zhang et al. [19] provided a review on applying multimodal fusion into clinical diagnosis and neuroscience research. Neuroimaging fusion can achieve higher temporal and spatial resolution, enhance contrast, correct imaging distortions, and bridge physiological and cognitive information. Wang and Zhang [20] proposed a new transfer-learning-based approach to identify multiple sclerosis more accurately. They used composite learning factor (CLF) to assign different learning factor to three types of layers. Four transfer learning settings were further tested and compared. A precomputation method was utilized to reduce the storage burden and accelerate the program. In a preliminary work by Huang [21], the authors collected long short-term memory (LSTM)-based models that can be used for sequence labeling, including LSTM, bidirectional LSTM, LSTM with a CRF layer, bidirectional LSTM respectively with a CRF layer, LSTM, Bi-LSTM, LSTM-CRF, and Bi-LSTM-CRF. These neural-based models (especially Bi-LSTM-CRF) achieved good robustness over conventional models. Carbonell et al. [22] proposed an end-to-end object detection network with branches to perform the handwritten text detection, transcription and named entity recognition at page level with a single feed-forward step. The proposed network can share features between different tasks. The results show that the model is capable of benefiting from shared features by simultaneously solving interdependent tasks. Kwob et al. [23] used the syllable bi-gram vector representation for Korean syllable-level named-entity recognition. They also proposed a novel model to make the joint vector representation of syllable bi-gram and Korean Eojeols positional information. The experiments showed that syllable-level named-entity recognition achieves not only good robustness but also faster than traditional morphological-level named-entity recognition by eliminating the morphological analysis process. Lee et al. [24] then proposed an integrated neural network model that consists of two layers of bidirectional gated recurrent unit models with conditional random field layers to perform morphological analysis and named entity recognition simultaneously. They used a two-phase training schema to train the entire framework. The proposed model can effectively alleviate the error propagation problem that frequently occurs in the pipeline architecture.

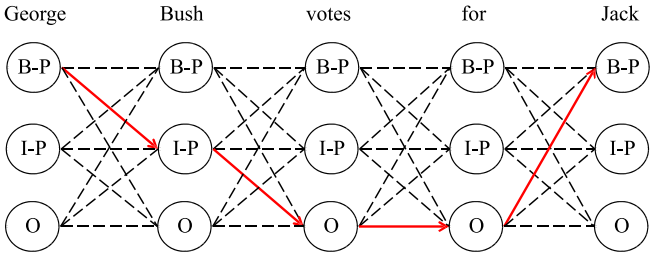

Fig. 2. The BIO encoding schema.

3. Preliminaries and problem statement

In this section, we give a brief overview of some background information as well as give a problem statement related to the work in this paper.

3.1. Conditional random fields with latent variables

Let us first consider a given sequence of what are known as observations $x = (x_1, \ldots, x_n)$. In CRF when dealing with variables (latent), our model first determines the method by which to assign sequence of labels $y = (y_1, \ldots, y_n)$ that come from a single finite set of labels Y. In place of modeling $P(y|x)$ in a direct manner, conventional conditional random fields sets latent variables h as an "inserted" set that is between both x and y making use of the well-known probabilistic chain rule as expressed in Eq. (1):

$$P(y|x) = \frac{1}{Z(x)} \sum_{h} P(y|h, x)P(h|x), \tag{1}$$

where $Z(x)$ can be used to denote normalization factor, h can be used to denote variable (latent), x can be used to denote sequence of the observations, and finally y can be used to represent sequence of the labels [25]. While we see that this model will allow the capturing of latent structure that exists between observations/labels, it is also better in other ways. Our models can find strong applications within the field of computer vision, taking into account gesture recognition from both sequence labeling as well as audio/video streams [26].

3.2. Encoding schema

Both the BIO as well as the BILOU encodings can represent clearly encoding schemas that are the mos tpopular in use today. BIO is clearly shown in Fig. 2, where **B** is beginning, **I** is inside, and finally **O** represents a given word that is not part of any segment. In Fig. 2, we describe that 'Michel' represents beginning of person, marked as **B-P**. 'Jordan' is inside of person marked as **I-P**. Next, the word 'would' is not part of any entity, as such marked with **O**. Furthermore, we show a much more complex scheme, known as BILOU, as shown through Fig. 3.

Through Fig. 3, we can denote **B** as the beginning, **I** as inside segment, which excludes the end word, **L** is the last word, and finally, **O** is any word that does not belong to any of the segments. As our example, we see that 'Michel' denotes the beginning of a person, marked as **B-P**. 'Jordan' is the last word of a person, marked then as **L-P**. Furthermore, 'would' does not belong to any entities, so it is marked with an **O**. Finally, 'Bush' is shown as the person entity with unit length, as such marked with **U-P** for a unit person. If we compare the sequence model not using an encoding schema, we see that many more features can be captured using encoding schemas, so clearly can have a defined positive impact on the performance of models.

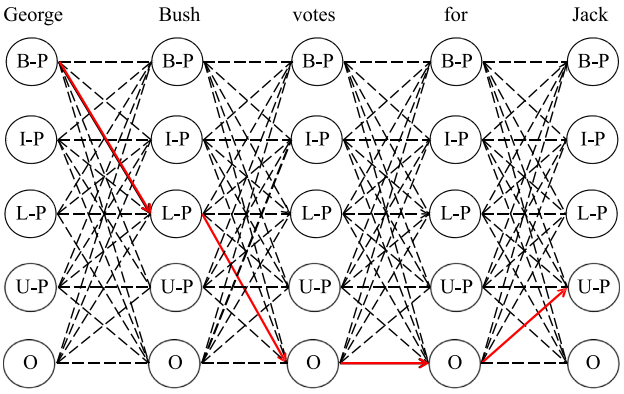

Fig. 3. The BILOU encoding schema.

3.3. Problem statement

Our problem statement can be formally defined as first taking into, consideration a given input sequence $x = (x_1, \ldots, x_k)$ which is of length k, as well as label of x which can be defined as tuple (u, y). This is defined as uth input word as associated with label y. A given label sequence then of x can be defined as in Eq. (2):

$$s = (s1, \ldots, sk) \tag{2}$$

where we see that $s_j = (u_j, y_j)$. We note here distinctly that the input sequence x as well as the label sequence s are of the same length. Therefore, if we are noting an input sequence x, then we can define the sequence labeling problem as that of finding label sequence s of x of the highest probability.

4. Proposed self-attention based conditional random fields with latent variables models

In this section, we introduce the neural CRF models dealing with latent variables. Sequence labeling models are commonly trained with supervised learning. Sequence labeling datasets are usually small. Thus, it is important to study how to enhance model performance without extra hand-crafted data and labels. In this paper, we take encoding schema as a latent variable for model training. To ensure a clear explanation, we introduce briefly conventional CRF, as well as present our proposed neural latent variable CRF. Finally, we explain the key differences between the models. Our first model as indicated earlier is SA-CRFLV-I, which can be defined as a sentence-level schema that can determine automatically the best encoding for use in sequence labeling. Secondly, our SA-CRFLV-II model which can be defined as a model at the word level hybrids both BIO and BILOU encoding schemes which were presented earlier. The hybrid nature of the second schema enhances prediction accuracy which will be shown later in our experimental work.

4.1. Conventional CRF

CRF is a popular model for sequence labeling. When directly compared with other well-known models, like MEM, CRF can incorporate many flexible features and be able to handle label biasing within the MEM model with strong results. When looking at the structure of conventional CRF free of any encoding schema, we see what is presented in Fig. 4. Here, P node is used to denote the person name of the entity node as well as O node is used to denote the non-entity node. From Fig. 4, we use dashed lines to encode every and every possible labeled path of any given input sequences. Through supervised training within the designed model, we see a labeled path (red line) in the CRF model. This red line

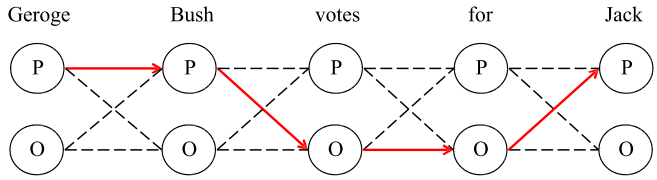

Fig. 4. Conventional CRF.

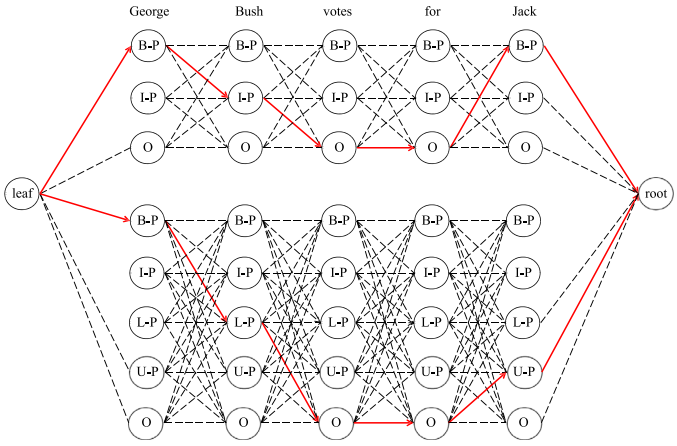

Fig. 5. The designed SA-CRFLV-I model.

corresponds to a distinct label. During training, model parameters are further optimized for the maximization of the probability of labeled paths. CRF model can provide a conditional probability of a potential output sequence s for input sequence given by x, as shown in Eq. (3):

$$p(s|x) = \frac{1}{Z(x)} exp\{W \cdot G(x,s)\}, \tag{3}$$

where $G(x,s)$ is used to denote feature function, W is used to denote weight vector, and finally $Z(x)$ is used to denote normalization factor. To be able to find best label sequence in CRF, we can let σ_j be used to denote best label sequence ends of jth input, denoted as (m,n,y) the label sequence starting at mth position, while ending at nth position, which is labeled as y. From this, σ_j is calculated recursively using Eq. (4).

$$\sigma_j = \max\Psi(j-1,j,y) + \sigma_{j-1}, \tag{4}$$

where $\Psi(j-1,j,y)$ is defined as feature value which can be defined over any label sequence denoted as $s = (j-l,j,y)$.

4.2. SA-CRFLV-I

When we compare LVCRF-I with classical CRF, it is observed that our proposed model can incorporate hidden variables which in turn allows the exploration of more information from input sequences. The performance of LVCRF-I largely depends on the handcraft features, which is similar to the classical CRF model. Thus, we incorporated a self-attention mechanism to automatically extract features. We introduce this model in two parts, we first introduce the SA-CRFLV-I and then the self-attention network. We show our designed CRF with the latent variable model clearly in Fig. 5.

As can be seen from Fig. 5, the model as proposed consists of exactly 2 parts. In the upper part, we include CRF alongside BIO schema, which was introduced earlier in Section 3.2. We define the connection relation as follows. Node **B** will be connected clearly to **I**, which indicates there exists an entity which can start at the current position and then continue to the next token, or in other words to **O** and then **B** nodes which indicates there exists an entity o which is of unit length at the current position. Node **I** can then be directly connected to **I**, which means there exists an entity that continues to the next token, or in other words to **O** and then on to **B** nodes, which means there exists at least one entity which ends at that node. Node **O** then can be directly connected on to nodes **B** and **O** respectively, which suggests clearly that no entity exists at current position. Since node **O** is not able to connect directly to node **I** due to the previously mentioned begging problem it must be labeled alongside **B**. We can see in the bottom portion of Fig. 5, there is corresponding imagery to CRF alongside BILOU schema. We describe the relationships as follows. Starting with node **B** which can be directly connected on to both nodes **I** and **L** respectively, which suggests there exists at least one entity which may start at the current position, on the other hand, nodes may not be directly connected to any of the nodes **O, B**, as well as **U**, due to that fact that any segment that has unit length must be labeled as **U**. Furthermore, node **I** may be connected as well to nodes **L** and **I** respectively because it denotes the inside of segment, therefore, it may not be directly connected to nodes **U, B**, as well as **O** which denote starting point of a new segment. We also see that node **L** may be connected to nodes **U, B**, and **O**, which means that the entity in question will end at the current position, however, may not be connected to nodes **I** and **L** respectively due to it denoting the end of a segment. The nodes **U** may not be directly connected to nodes **U, O**, and **B**, which suggests that there exists some entity that has a unit length which is at the current position and may not be able to directly connect to nodes **L** and **I** respectively because there should exist node **B** which denotes the start of the segment before them. It is suggested that there does not exist an entity at current position because node **O** may be connected on to nodes **U, O**, and **B**, however can not connect to nodes **L** and **I** due to node **B** being before them. We also show the leaf node which is in the left portion of Fig. 5 which denotes the start of a given sentence, as well as the root node in the right portion which represents the ending of the sentence.

In Fig. 5 we give an input sequence and show how our graph model will provide 2 separate labeled paths. In other words, a path which corresponds to BIO schema as well as a path which corresponds to BILOU schema, respectively.

The performance of the conventional CRF based models largely depends on the hand-craft features. For different natural language processing tasks and datasets, it thus requires different feature engineering, which is time-consuming. Thus, we combined the proposed models with neural networks to automatically extract features for LVCRF-I models. This neural network utilizes the Bi-LSTM to compute the vector representation for each word. Then we concatenate the forward and backward direction of the Bi-LSTM to form the vector representation of each token. This permits the tokens to be sensitive to the contexts where they occur and is typical of neural network sequence prediction models. The self-attention mechanism is similar to as [27] to further compute the vector representation of each time-step, and the formulation is defined as follows:

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \tag{5}$$

where Q, K and V are input vector representations. A fully connected neural network is utilized as the projection layer to compute the neural feature scores. The formulation is defined as follows:

$$features_{neural}(i) = softmax(\theta x_i + \beta), \tag{6}$$

where x_i is the vector representation of the i-th timestep, θ and β is the learnable parameters. Different from LVCRF models, SA-CRFLV-I utilizes self-attention neural networks to compute features for the CRF layer. As described before, there are $N+4$ edges connected to a B node, $N+5$ edges connected to a I node, $3N+1$

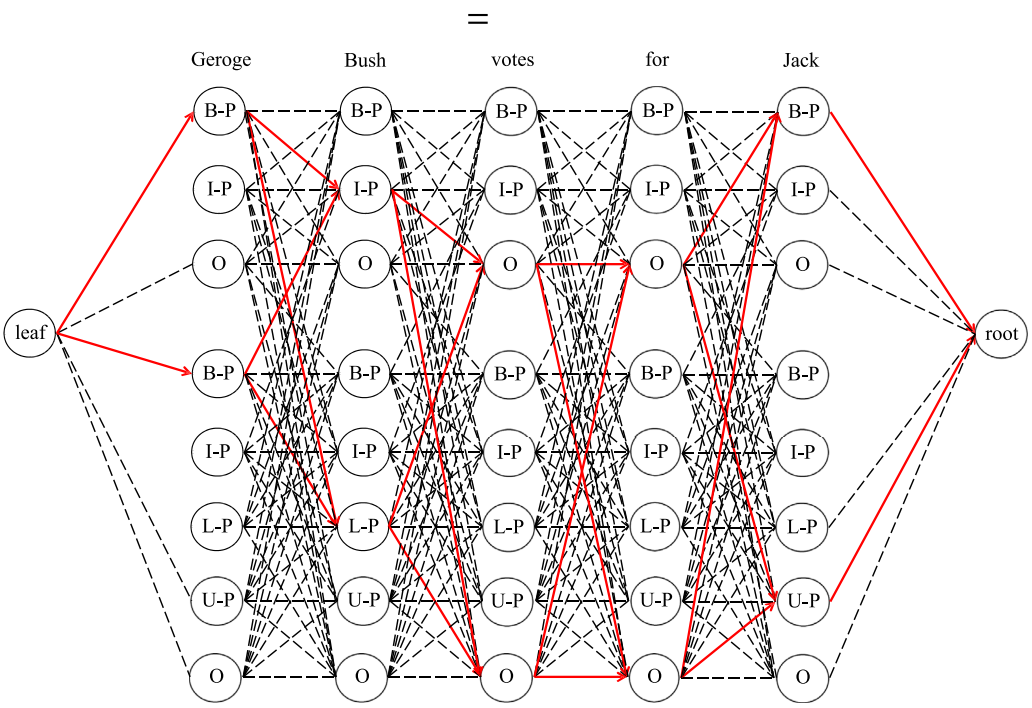

Fig. 6. The designed SA-CRFLV-II.

edges connected to a O node, $2N+1$ edges connected to a L node and $2N+1$ nodes connected to a U node. Thus, the output of the self-attention network of each timestep is a $6N+8$ dimension vector. For every existing edge in the CRF model, the self-attention network outputs a score (i.e., neural features) which will be computed together with hand-craft features in CRF models. Thus the designed model achieves good robustness over different datasets and sequence labeling tasks since the neural features are based on the specific datasets and tasks. The LVCRF model only depends on the hand-craft features, which is hard to be generalized for different datasets and tasks. Compared with conventional LSTM models, the attention mechanism can accelerate the model training since all timesteps can be processed in parallel. The attention mechanism also achieves good robustness when it is utilized in AI models since it can automatically generate weights for each timestep. The disadvantages of the attention mechanism are that it requires more computing resources to perform the training progress especially when a sequence is very long.

4.3. SA-CRFLV-II

SA-CRFLV-I is shown to be and designed to be a model at the sentence level because it chooses encoding schema directly for each sentence. When we compare this fact of SA-CRFLV-I with SA-CRFLV-II, we see that the latter represents a model at the word-level because it chooses encoding schema automatically for every word. We give a detailed account of SA-CRFLV-II in Fig. 6, where the two parts of the model are shown.

SA-CRFLV-II is shown to have a similar structure when compared with SA-CRFLV-I. The main difference is edges that connect upper CRF and bottom CRF. As an example, upper node **B** c may be connected on to bottom node **I**, which means that there is at least one entity using BIO schema that has current starting position then continues to next token using BILOU schema, or in other words on to node **L** which is in bottom CRF. This denotes that there is at least one entity that is of unit length at the current position using the BILOU schema. The main difference that exists when comparing SA-CRFLV-I to SA-CRFLV-II is the latter will allow the transformation of encoding schema when given a certain sentence of input.

Considering an example through Fig. 6, for a given sequence of input, we can say that there are 2^n potential labeled paths, and let n denote the sentence length. Through the set theory, we can say that all paths have equal probability. In other words, they may be able to label as “George Bush” as well as “Jack” which are both named entities and then showed that “votes for” is not a named entity, or also called a non-named entity. Through decoding progress, our proposed model can provide red lines as a subset, where we say that the lines are connected end-to-end to one another. Through this, the resulting part is that every word may be labeled with the use of different schemas for encoding. In other words, “George” may be labeled with nodes **B-P** making use of the `BIO` schema, whereas “Bush” may be labeled with nodes **L-P** using `BILOU` schema.

4.4. Training, inference, and decoding

Making use of CRF, a log-linear approach is adopted for our objective function, which can be expressed as in Eq. (7).

$$L(w) = \sum_i log \sum_{y'} exp(w^T f(x_i, y')) - \sum_i w^T f(x_i, y) + \lambda w^T w, \quad (7)$$

where we can define (x_i, y_i) as sentence x_i as well as labeled path y_i as the correct one,. Furthermore, last term is used to represent a $L2$ regularization term using a λ value of 0.01. The objective function may be optimized using standard gradient-based methods.

More importantly, for any given input sentence x, the probability of predicting some potential output sequence y can be expressed as in Eq. (8).

$$p(y|x) = \frac{exp(w^T f(x, y))}{\sum_{y'} exp(w^T f(x, y'))}, \quad (8)$$

where here we let $f(x, y)$ denote the feature vector which is defined over input-output pair (x, y) as well as the weight vector w which provides model parameters.

We make use of what is known as an inside-outside algorithm which is similar to what is given in Muis and Lu [16] for the inference process. First, the inference algorithm is used to calculate the inside score for every node from leaf to root. Next, the outside score is calculated from root to leaf. Furthermore, the internal score can be calculated just by summing features scores that are associated with edge linking of the current node as well as its child nodes, at the same time we let the internal score be calculated using the bottom-up (left-2-right) dynamic programming process. We define path score as the product of inside score which is stored at the child node as well as feature score which is defined over a given edge that is allowing them to connect. For the computation of outside score we use a similar methodology but from right-2-left this time.

Both inside as well as the outside score for any given step may be calculated with known time complexity of $\mathcal{O}(N^2)$, where we let N denote # of entity types. This is because every node may be connected to a maximum of $2N$ total nodes (2/encoding schema total N schemas). Furthermore, there are $2N$ nodes given at every time step as defined by $2N * 2N = O(N^2)$. Therefore, for any given input sentence that has length T as well as N entity types, we can say that the time complexity for our model is $\mathcal{O}(TN^2)$. This time complexity is similar to conventional CRF models. Using the Viterbi decoding algorithm which using dynamic programming, we can obtain an output path of high probability. Our training model is very similar to what would be considered in the conventional graphic model, known as forward-backward algorithms. We show this algorithm in Algorithm 1.

Algorithm 1 forward-backward algorithm.

```
for each epoch do
    for each batch do
        1) Self-attention based model forward pass thus calculate the features
        2) LVCRF-I/II forward and backward pass thus calculate the inside and outside score
        3) Self-attention based model backward pass thus calculate gradient
        4) Perform the updating model for parameters
    end for
end for
```

4.5. Features

In this section, CRF features that are used to compute **G(x,s)** as shown through Eq. (3) is shown. More specifically, input features as follows are used.

- **Word features:** the window size is set as 3 for the words appearing around the current position.
- **POS tag features (if available):** the window size is set as 3 for the POS tags appearing around the current position.
- **Word *n*-gram features:** the *n*-gram size is respectively set as 2, 3, and 4 of the current position.
- **POS *n*-gram features (if available):** the settings are the same as Word *n*-gram features.

5. Experimental evaluation

We test our proposed two models named SA-CRFLV-I and SA-CRFLV-II on four tasks regarding natural language processing (NLP) such as NER, chunking, reference parsing, and tagging task in POS. The experimental results of the developed two models are then compared to the conventional CRF with two different schemas for encoding. As the input of the SA-CRFLV-I/II, a task-specific pre-trained word embedding (64 dimensions) is used. The CRF-BIO is the traditional CRF with the BIO encoding scheme while the CRF-BILOU is the traditional CRF with the BILOU encoding scheme. In comparison, the models use the same functionality as described in 4.5 section.[1]

Table 1
Corpora Statistics of the used datasets.

Name	Task	# train	# labels	# dev	# test
CoNLL2003	NER	14,987	8	3466	3684
BC2GM	NER	12,500	3	2500	5000
CoNLL2000	Chunking	8936	22	N/A	2012
Cora	Ref parsing	500	13	N/A	N/A
PTB POS	Pos tagging	39,831	45	1699	2415

Table 2
Performance for different algorithms in ConLL2003 for the NER task.

Compared models	Recall	Precision	F1
CRF-BIO	83.59	84.10	83.84
CRF-BILOU	84.36	83.82	84.09
LSTM-CRF	91.38	90.21	90.79
LVCRF-I	84.71	84.19	84.46
LVCRF-II	85.05	84.15	84.59
SA-CRFLV-I	93.01	90.78	91.88
SA-CRFLV-II	92.88	91.32	**92.09**

5.1. Datasets

In the experiments, five standard databases regarding four different tasks such as NER, chunking, ref parsing, and POS tagging are conducted for evaluation to show the results of the compared models. The corporate statistics of the four different tasks in five datasets are thus illustrated in Table 1.

Here, the ConLL2003 [28] is to verify the NER task having four various name entities as Person, Location, Organization, and Misc. The BC2GM is also the NER task that is the BioCreative II Gene Mention corpus having 20,000 sentences extracted from the abstract part of the published articles in biomedical fields. It was annotated using a single NE class for the names of genes, proteins, and related entities. ConLL2000 [29] belongs to the chunking task, and Sections 15–18 from the Penn Treebank of the Wall Street Journal are then considered as the training data, and section 20 is then used as the testing validation. Cora [30] is a reference parsing task with 13 labeled fields having 500 reference strings (i.e., author, title, journal title, vol., page, date, among others). Penn TreeBank (PTB) POS is also the reference parsing task with 45 labeled fields having 30,000 sentences. Results for different tasks are then given below.

5.2. NER tasks

Tables 2 and 3 shows the experimental results for two NER task regarding CoNLL2003 and BC2GM datasets. The best results are then marked with an underline. As previously stated, the LVCRF-I can be seen as a mixture of the CRF-BIO and CRF-BILOU, which is why its performance was robust and outperformed both the CRF-BIO and the CRF-BILOU. This result proved that the CRFLV-I could automatically identify the best encoding scheme by the input sentence. The proposed SA-CRFLV-I and SA-CRFLV-II have better performance than that of LVCRF-I and LVCRF-II. For example, the SA-CRFLV-II achieved the best results among all compared algorithms.

[1] The developing tool StatNLP can be found in https://statnlp-research.github.io/

Table 3
Performance for different algorithms in BC2GM for the NER task.

Compared models	Recall	Precision	F1
CRF-BIO	87.88	86.5	87.18
CRF-BILOU	88.05	86.88	87.46
LSTM-CRF	90.36	89.45	89.90
LVCRF-I	89.25	86.81	88.01
LVCRF-II	89.39	86.78	88.06
SA-CRFLV-I	91.20	90.71	90.95
SA-CRFLV-II	91.62	90.87	**91.24**

Table 4
Performance for different algorithms in CoNLL2000 for the chunking task.

Compared models	Recall	Precision	F1
CRF-BIO	89.89	90.15	90.01
CRF-BILOU	89.88	90.05	89.96
LSTM-CRF	90.78	92.34	91.55
LVCRF-I	90.23	90.12	90.17
LVCRF-II	90.41	90.08	90.24
SA-CRFLV-I	91.78	93.11	92.44
SA-CRFLV-II	92.32	93.01	**92.66**

Table 5
Performance for different algorithms in Cora for the reference parsing task.

Compared models	Recall	Precision	F1
CRF-BIO	80.61	77.92	79.24
CRF-BILOU	81.21	78.35	79.75
LSTM-CRF	82.01	78.99	80.47
LVCRF-I	81.56	78.15	79.81
LVCRF-II	81.89	78.25	80.02
SA-CRFLV-I	81.94	79.62	80.76
SA-CRFLV-II	82.23	79.41	**80.79**

Table 6
Performance for different algorithms in PTB POS for the pos tagging task.

Compared models	Recall	Precision	F1
CRF-BIO	95.99	93.41	94.68
CRF-BILOU	95.27	93.45	94.35
LSTM-CRF	95.37	96.10	95.73
LVCRF-I	95.51	94.22	94.86
LVCRF-II	95.19	94.71	94.95
SA-CRFLV-I	96.32	96.20	96.25
SA-CRFLV-II	96.24	96.33	**96.28**

For different datasets, SA-CRFLV-I and SA-CRFLV-II achieve better robustness than that of LVCRF-I and LVCRF-II since the attention-based networks can extract dataset-specific features, while LVCRF-I and LVCRF-III can only use hand-craft features. Throughout this study, we are proposing a CRF system which uses the encoding scheme as a latent variable. The result showed that in a neural-based model, the proposed structure could easily outperform the CRF model, as both an independent variable and an embedded layer. As predicted the SA-CRFLV-II performance was marginally better than the SA-CRFLV-I performance. The CRF-BIO and CRF-BILOU showed low results, so is the CRF's output with different encoding schemas.

5.3. Chunking

In the chunking task on the CoNLL2000 dataset [29], Table 4 compares the results of different algorithms. As shown in Table 4, the proposed models exceeded the baseline CRF-BIO, CRF-BILOU, and LVCRF models. The developed SA-CRFLV-I and SA-CRFLV-II have achieved a similar performance in this task. The CRF with the BIO encoding scheme performed better on the chunking task, while the CRF with the BILOU encoding scheme was better on the NER task. This is reasonable since none of the encoding schemes was the best for all cases and applications, so it was necessary to use different encoding schemes for the different input sentences that we proposed in this paper. The SA-CRFLV-II also achieved the best performance among all models on the CoNLL2000 dataset.

5.4. Reference parsing

In comparison with chunking and NER tasks, the reference parsing gives more segmental information. Table 5 shows the compared results of different models in reference parsing task of the Cora dataset [30]. From the results, it can be seen that the CRF-BILOU exceeds the CRF-BIO for two comparable basic models, i.e., the CRF-BIOU and the CRF-BILOU. The reason could be possibly referred to as that the CRF-BILOU was able to capture more segmental information. Thus, boundary words are considered an important term in this task. The two proposed models achieve great robustness since their performance was superior to the CRF-BILOU and LVCRF models.

5.5. POS tagging

POS Tagging is the task of assigning each word with a syntactic tag to an input sentence. Compared to the three above tasks such as NER, chunking, and reference parsing, the POS label has less information on the segment level. However, the performance of the two proposed models was still higher than that of the CRF and LVCRF models, which can be seen in Table 6. For example, the designed SA-CRFLV-II achieves the best results in terms of F1 compared to the other models. Thus, we can conclude that the designed two models outperform any of the existing models in different tasks and applications.

6. Conclusion

In this paper, we study the problem of sequence labeling which may often be used as a step of pre-processing for NLP. Sequence labeling can help many computers and machines get a better understanding of a sequence of text. Our in-depth evaluation using different schemas for encoding led to the design and introduction of 2 novel neural CRFs with latent variables which are shown to improve the performance of sequence labeling. At the sentence level and the word level respectively, CRFLV-I, as well as CRFLV-II, showed strong performance through experimental evaluation. In this work, two encoding schemas are considered and combined as a latent variable. In future works, another encoding schema can be further explored such as BILO encoding schema. Taking more encoding schemas as latent variables can be a feasible way to enhance model performance. Different neural network structures and attention mechanisms should also be further explored.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Ding, S. Qu, Y. Xi, S. Wan, Stimulus-driven and concept-driven analysis for image caption generation, Neurocomputing 398 (2020) 520–530.
[2] C. Ma, C. Shen, A. Dick, Q. Wu, P. Wang, A. van den Hengel, I. Reid, Visual question answering with memory-augmented networks, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 6975–6984.

[3] L. Ratinov, D. Roth, Design challenges and misconceptions in named entity recognition, in: The Conference on Computational Natural Language Learning, 2009, pp. 147–155.
[4] Y. Xi, Y. Zhang, S. Ding, S. Wan, Visual question answering model based on visual relationship detection, Signal Process. 80 (2020) 115648.
[5] Y.D. Zhang, Z. Dong, S.H. Wang, X. Yu, X. Yao, Q. Zhou, H. Hu, M. Li, C. Jiménez-Mesa, J. Ramirez, et al., Advances in multimodal data fusion in neuroimaging: overview, challenges, and novel orientation, Inf. Fusion 64 (2020) 149–187.
[6] S.H. Wang, V.V. Govindaraj, J.M. Górriz, X. Zhang, Y.D. Zhang, COVID-19 classification by FGCNet with deep feature fusion from graph convolutional network and convolutional neural network, Inf. Fusion 67 (2021) 208–229.
[7] L.E. Baum, T. Petrie, Statistical inference for probabilistic functions of finite state Markov chains, Ann. Math. Stat. 37 (6) (1966) 1554–1563.
[8] L.E. Baum, J.A. Eagon, An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology, Bull. Am. Math. Soc. 37 (3) (1967) 360–363.
[9] L.E. Baum, An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process, Inequalities 3 (1972) 1–8.
[10] J.D. Lafferty, A. Mccallum, F.C.N. Pereira, Conditional random fields: probabilistic models for segmenting and labeling sequence data, in: The Eighteenth International Conference on Machine Learning, 2001, pp. 282–289.
[11] S. Sarawagi, W.W. Cohen, Semi-Markov conditional random fields for information extraction, in: The Neural Information Processing Systems, 2004, pp. 1185–1192.
[12] A.L. Berger, S.A.D. Pietra, V.J.D. Pietra, A maximum entropy approach to natural language processing, Comput. Linguist. 22 (1) (1996) 39–71.
[13] H. Tseng, P. Chang, G. Andrew, D. Jurafsky, C. Manning, Sequential labeling with latent variables, in: The Workshop on Chinese Language Processing, 2015, pp. 168–171.
[14] H. Zhao, C.N. Huang, M. Li, T. Kudo, An improved chinese word segmentation system with conditional random field, in: The Workshop on Chinese Language Processing, 2006, pp. 162–165.
[15] N.V. Cuong, N. Ye, W.S. Lee, L.C. Hai, Conditional random field with high-order dependencies for sequence labeling and segmentation, J. Mach. Learn. Res. 15 (1) (2014) 981–1009.
[16] A.O. Muis, W. Lu, Weak semi-Markov CRFs for noun phrase Chunking in informal text, in: The North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 714–719.
[17] J.C.W. Lin, Y. Shao, J. Zhang, U. Yun, Enhanced sequence labeling based on latent variable conditional random fields, Neurocomputing 403 (2020) 431–440.
[18] J.C.W. Lin, Y. Shao, Y. Djenourid, U. Yun, ASRNN: a recurrent neural network with an attention model for sequence labeling, Knowl. Based Syt. 212 (2021) 106548.
[19] Y.-D. Zhang, Z. Dong, S.-H. Wang, X. Yu, X. Yao, Q. Zhou, H. Hu, M. Li, C. Jiménez-Mesa, J. Ramirez, et al., Advances in multimodal data fusion in neuroimaging: overview, challenges, and novel orientation, Inf. Fusion 64 (2020) 149–187.
[20] S.H. Wang, Y.D. Zhang, Densenet-201-based deep neural network with composite learning factor and precomputation for multiple sclerosis classification, ACM Trans. Multimed. Comput. Commun. Appl. 16 (2s) (2020) 1–19.
[21] Z. Huang, W. Xu, K. Yu, Bidirectional LSTM-CRF models for sequence tagging, arXiv:1508.01991s2015.
[22] M. Carbonell, A. Fornés, M. Villegas, J. Lladós, A neural model for text localization, transcription and named entity recognition in full pages, Pattern Recognit. Lett. 136 (2020) 219–227.
[23] S. Kwon, Y. Ko, J. Seo, Effective vector representation for the korean named-entity recognition, Pattern Recognit. Lett. 117 (2019) 52–57.
[24] H.-g. Lee, G. Park, H. Kim, Effective integration of morphological analysis and named entity recognition based on a recurrent neural network, Pattern Recognit. Lett. 112 (2018) 361–365.
[25] Y. Zhao, H. Li, S. Wan, A. Sekuboyina, X. Hu, G. Tetteh, M. Piraud, B. Menze, Knowledge-aided convolutional neural network for small organ segmentation, IEEE J. Biomed. Health Inform. 23 (4) (2019) 1363–1373.
[26] S. Wan, Y. Xia, L. Qi, Y.-H. Yang, M. Atiquzzaman, Automated colorization of a grayscale image with seed points propagation, IEEE Trans. Multimed. 22 (7) (2020) 1756–1768.
[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[28] D. Yu, L. Deng, A.A. Acero, Using continuous features in the maximum entropy model, Pattern Recognit. Lett. 30 (14) (2009) 1295–1300.
[29] D. Okanohara, Y. Miyao, Y. Tsuruoka, J. Tisuji, Improving the scalability of semi–Markov conditional random fields for named entity recognition, in: The Annual Meeting of the Association for Computational Linguistics, 2006, pp. 465–472.
[30] L.E. Baum, G.R. Sell, Growth transformations for functions on manifolds, Pac. J. Math. 27 (2) (1968) 211–227.