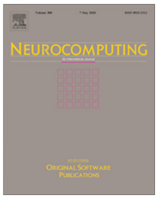
Neurocomputing 403 (2020) 431–440

Contents lists available at ScienceDirect

Neurocomputing

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



Enhanced sequence labeling based on latent variable conditional random fields

Jerry Chun-Wei Lin [a,*], Yinan Shao [b], Ji Zhang [c], Unil Yun [d]

[a] Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway
[b] Alibaba Inc., Hangzhou, Zhejiang, China
[c] School of Sciences, University of Southern Queensland, Australia
[d] Department of Computer Engineering, Sejong University, Republic of Korea

ARTICLE INFO

Article history:
Received 3 September 2019
Revised 4 April 2020
Accepted 29 April 2020
Available online 8 May 2020
Communicated by Shiliang Sun

Keywords:
Latent CRF
Sequence labeling
Encoding schema
Natural language processing

ABSTRACT

Natural language processing is a useful processing technique of language data, such as text and speech. Sequence labeling represents the upstream task of many natural language processing tasks, such as machine translation, text classification, and sentiment classification. In this paper, the focus is on the sequence labeling task, in which semantic labels are assigned to each unit of a given input sequence. Two frameworks of latent variable conditional random fields (CRF) models (called LVCRF-I and LVCRF-II) are proposed, which use the encoding schema as a latent variable to capture the latent structure of the hidden variables and the observed data. Among the two designed models, the LVCRF-I model focuses on the sentence level, while the LVCRF-II works in the word level, to choose the best encoding schema for a given input sequence automatically without handcraft features. In the experiments, the two proposed models are verified by four sequence prediction tasks, including named entity recognition (NER), chunking, reference parsing and POS tagging. The proposed frameworks achieve better performance without using other handcraft features than the conventional CRF model. Moreover, these designed frameworks can be viewed as a substitution of the conventional CRF models. In the commonly used LSTM-CRF models, the CRF layer can be replaced with our proposed framework as they use the same training and inference procedure. The experimental results show that the proposed models exhibit latent variable and provide competitive and robust performance on all three sequence prediction tasks.

1. Introduction

Sequence labeling is often the first step in text data processing. Sequence labeling represents the task of identifying and assigning a semantic label to each unit/subsequence of the input sequences. It can help machines better understand the components or structure of the given contexts. Conventionally, sequence labeling is achieved by named entity recognition that extracts name entities (i.e., person name, company name, etc.) from the text, chunking that identifies the constituent parts of sentences (i.e., nouns, verbs, adjectives, etc.), and reference parsing that extracts the information (i.e., author, title, journal, etc.) from a given reference string. Sequence prediction conducts fundamental research in natural language processing tasks. Due to its importance for the downstream tasks, including the relation extraction [1,2], entity linking [3], and co-reference resolution [4], it has received substantial attention in recent decades. The conventional sequence labeling models, such as conditional random fields (CRF) and the maximum entropy model (MEM), study the conditional probability over the input sequence by representing the input unit, i.e., characters or words. The segmentation models, e.g., the semi-Markov random fields (semi-CRF), represent the text span of the input sequence (i.e., subsequence) directly. In Refs. [5,6], it was shown that the encoding schema affected model performance. The BIO and BILOU represent the most popular encoding schemas, where **B** stands for beginning, **I** stands for inside, **O** stands for outside, **L** stands for last, and **U** stands for unit.

An example of using a different encoding schema on a sentence is presented in Fig. 1. In Fig. 1, 'Michel' represents the beginning of a person entity, and is marked with B in both encoding schemas. However, 'Jordan' denotes inside a person entity in the BIO, and thus is marked with I, whereas it is the last word of a person entity in the BILOU, and thus, is marked with L. The word 'Bush' is marked

* Corresponding author.
E-mail addresses: jerrylin@ieee.org (J.C.-W. Lin), Ji.Zhang@usq.edu.au (J. Zhang), yunei@sejong.ac.kr (U. Yun).

https://doi.org/10.1016/j.neucom.2020.04.102
0925-2312/© 2020 The Authors. Published by Elsevier B.V.

BIO encoding	Michel B-PER	Jordan I-PER	would O	choose O	Bush B-PER
BILOU encoding	Michel B-PER	Jordan L-PER	would O	choose O	Bush U-PER

Fig. 1. The BIO and BILOU encoding schemas.

with B in the BIO encoding schema because it represents the beginning of a person entity, and it is marked with U in the BILOU encoding schema wherein it denotes a unit length person entity.

Different encoding schemas can lead to different performance on different models and sequence-labeling tasks. For a given model and a sequence-labeling task, it is common to use the validation set for obtaining the best encoding schema, which is a non-trivial task. Accordingly, in this paper, two latent variable CRFs, which can automatically choose the best encoding scheme for a given input sentence, are proposed. In the first designed CRF-based model, called LVCRF-I, the input sentence can be labeled by two encoding schemas simultaneously, whereby optimizing the parameters to maximize the probability of both encoding schemas. The second designed model, called LVCRF-II, chooses the encoding schema on a word-level rather than the sentence-level that hybrids the path in two encoding schemas. The first advantage of the proposed models is that the developed models use a validation set to choose the encoding schema manually, which can be easily and automatically performed by our latent variable CRFs. The second advantage of the developed models is that the accuracy can be greatly improved using any other information (i.e., feature engineering task) because the models choose the most suitable encoding schema for each sentence rather than adopting one particular encoding schema for all the sentences. The main contributions of this work are as follows.

- Two latent variable CRFs (LVCRF-I and LVCRF-II) for the sequence labeling are proposed. Both CRFs can be applied to different sequence labeling subtasks, including the part-of-speech tagging, named entity recognition, chunking, and others.
- In the proposed models, the encoding schema is used as latent variables to capture the structures of hidden variables and observed data. The first model can choose the best encoding schema for the whole input sentence automatically, while the second model can determine the best encoding schema for every word in the input sentence.
- The two proposed models use different encoding schemas, as a latent variable in the conventional CRF in two ways. The two model frameworks can also be used in other CRF-based models.
- Empirically, it is shown that choosing the best encoding schema has a stable impact on the performance. The performance of the proposed latent variable model is much better than the conventional CRF with the BIO or BILOU encoding schema.

2. Literature review

The traditional mention-extraction models include the hidden Markov model (HMM) [7–9], max-entropy model (MEM) [10], conditional random field model (CRF) [11], and semi-Markov random field model (semi-CRF) [12]. These models are linear models that can capture correlations between neighboring labels and jointly decode the best chain of labels for a given input sequence. Baum and coworkers [7–9] proposed a hidden Markov model (HMM) that can be represented as a dynamic Bayesian network. Fine et al. [13] proposed a hierarchical hidden Markov model (HHMM), which represents a recursive hierarchical generalization of the vanilla hidden Markov model. Zhang et al. [14] built the ICTCLAS system, which uses a hierarchical hidden Markov model to incorporate the segmentation of Chinese words, part-of-speech tagging, disambiguation, and unknown-word recognition in a comprehensive theoretical frame. Shen et al. [15] proposed a general hidden Markov model-based named-entity recognizer in the biomedical domain. Berge et al. [10] pioneered a maximum entropy model (MEM) in natural language processing. McCallum et al. [16] proposed a maximum entropy Markov model (MEMM), which represents a graphical model for sequence labeling, and it combines features of both the HMM and the MEM. Yu et al. [17] investigated the problem of using continuous features in the MEM. They explained why the MEM with the moment constraint (MEMC) worked well with binary features but not with continuous features. Ratnaparkhi [18] proposed a statistical model that was trained with a corpus annotated, including the part-of-speech tags, and assigned them to previously unseen text with high accuracy. The above work demonstrated the effectiveness of specialized features in modeling difficult tagging decisions and proposed a training strategy that mitigated the corpus-consistency problems discovered during the implementation of specialized features. Rosenberg et al. [19] proposed the mixture-of-parents maximum entropy Markov model (MoP-MEMM). This model allows tractable incorporation of long-range dependencies between nodes by restricting the conditional distribution of each node to a mixture of parent distributions.

Conditional random field (CRF) models were proposed by Lafferty et al. [11]. These models denote a class of statistical modeling method that has been often applied to solve the sequence-prediction problems. These models have several advantages over hidden Markov models and stochastic grammars for such tasks, including the ability to relax strong independence assumptions made in these models. Tseng et al. [20] presented a Chinese word segmentation (CWS) system based on the CRF models. Zhao et al. [21] considered the CWS problem as a character-based tagging problem under a conditional random field framework. Instead of considering a method focused only on a feature template as in the previous work, they considered both feature-template and tag-set selection. They demonstrated a significant performance difference using selected tag sets. Cuong et al. [22] considered the problem of incorporating high-order dependencies between labels or segments in the conditional random field. The Semi-Markov conditional random field (semi-CRF) model was proposed by Sarawagi and Cohen [12]. Importantly, features of the semi-CRFs can measure properties of segments, and transitions within a segment can be non-Markovian. Okanohara et al. [23] presented techniques to apply the semi-CRFs to the named entity-recognition tasks with a tractable computational cost. Nguyen et al. [24] extended the first-order semi-CRFs to include higher-order semi-Markov features and proposed efficient inference and learning algorithms under the assumption that the higher-order semi-Markov features were sparse. Muis and Lu [25] proposed the weak semi-Markov conditional random field for noun-phrase chunking. In the conventional semi-CRF, the model intuitively decides the length and type of the next segment simultaneously, while in the weak semi-CRF, the model tries to propose a weaker variant that makes these two decisions separately by restricting each node to connect to either only the nodes of the same label in the next segment or to all the nodes in the next word. The weak semi-CRF model yields performance similar to that of the conventional semi-CRFs, but runs significantly faster.

The deep learning-based methods show advantages in the sequence labeling task. Huang et al. [26] proposed a variety of the long short-term memory (LSTM)-based models for sequence labeling, including the LSTM networks, the bidirectional LSTM (Bi-LSTM) networks, the LSTM with a CRF layer (LSTM-CRF), and the bidirectional LSTM with a CRF layer (Bi-LSTM-CRF). Their model achieved the state-of-the-art accuracy on the POS, chunking, and NER datasets, and the performance was less dependent on the word embedding than on previous observations. Liu et al. [27] proposed a neural semi-Markov conditional random field, which composes the embedding of both input units and segments. They conducted the experiments with the named entity recognition (NER) and Chinese word segmentation (CWS) tasks. Ma and Hovy [28] proposed a CNN-LSTM-CRF model that benefits from both word- and character-level representations. Their model is the end-to-end model, and it does not require feature engineering or data pre-processing. Rei et al. [29] used the character-level information to address the out-of-vocabulary (OOV) issue in sequence labeling. They investigated the character-level extensions of the conventional LSTM-CRF structure models. The encoded character-level information was combined with the pre-trained word embeddings using the attention mechanism, enabling the model to decide how much information to use from a word- or character-level component.

The sequence labeling has been previously achieved using different approaches based on the latent variable models. Sun and Nan [30] proposed a latent discriminative model called the Latent Semi-CRF, which incorporates advantages of two modeling approaches, i.e., the latent dynamic CRF and the semi-CRF, that model the sub-structure of a class sequence and learn dynamics between the class labels for detecting the Chinese base-phrases. Petrov and Dan [31] introduced a discriminative latent variable approach for syntactic parsing in which rules exist at multiple scales of refinement. Such a model is formally a latent variable CRF grammar over trees learned by iteratively splitting grammar productions. Sun et al. [32] proposed a latent semi-CRF model to detect the new words and their POS synchronously regardless of the type of new words from the Chinese text without using the pre-segmentation process. Sun and Tsujii [33] described the latent-dynamic inference (LDI), which produces the optimal label sequence of the latent conditional models by using efficient search strategy and dynamic programming. Sun et al. [34] combined multi-view CRF learning by utilizing consensus and complementary principles for sequence labeling. It uses different neural networks for feature extraction from multiple views. A joint representation space for the retrieved features thus the minimal distance between two views for regularization can thus be achieved.

3. Preliminaries and Problem Statement

This section briefly introduces the preliminaries and problem statement of this research work.

3.1. Latent variable CRF

Consider a sequence of observations $x = (x_1, \ldots, x_n)$. In the latent variable CRF, the model determines how to assign a sequence of labels $y = (y_1, \ldots, y_n)$, from one finite set of labels Y. Instead of modeling $P(y|x)$ directly, as the conventional CRF does, a set of latent variables h is "inserted" between x and y using the chain rule of probability, which is expressed as,

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

where $Z(x)$ denotes the normalization factor, h denotes the latent variable, x denotes the sequence of observations, and y represents the sequence of labels. This model allows capturing the latent structure between observations and labels. These models find applications in the computer vision field, especially in the gesture recognition from video streams and sequence labeling.

3.2. Encoding Schema

The BIO and BILOU encodings represent the most popular encoding schemas. The BIO encoding schema is presented in Fig. 2, where **B** denotes the beginning of a segment, **I** represents the inside of a segment, including the ending word, and **O** stands for the word that does not belong to any segment. As shown in Fig. 2, 'Michel' represents the beginning word of the person entity, so it is marked with **B-P** (Begin-Person). 'Jordan' denotes the inside word of the person entity, so it is marked with **I-P** (Inside-Person). The word 'would' does not belong to any entity, so it is marked with **O**. A more complex schema, called BILOU, is shown in Fig. 3.

In Fig. 3, **B** denotes the beginning of a segment, **I** denotes the inside of a segment, excluding the ending word, **L** denotes the last word of a segment, and **O** stands for the word that does not belong to any segment. As shown in Fig. 3, 'Michel' denotes the beginning word of the person entity, so it is marked with **B-P** (Begin-Person). 'Jordan' denotes the last word of the person entity, so it is marked with **L-P** (Last-Person). The word 'would' does not belong to any entity, so it is marked with **O**. The word 'Bush' denotes the person entity with a unit length, so it is marked with **U-P** (Unit-Person). Compared with the sequence model without any encoding schema, more features can be captured by the encoding schema, so it can exert a positive impact on the model performance.

3.3. Problem Statement

Formally, considering an input sequence x = $(x_1, \ldots, x_k)$ of a length k, a label of x is defined as a tuple (u, y), which means the u-th input word is associated with a label y. A label sequence of x is defined as s = $(s1, \ldots, sk)$, where s_j= (u_j, y_j). It should be noted that the input sequence x and the label sequence s have the same length. Given an input sequence x, the sequence labeling problem can be defined as the problem of finding the most probable label sequence s of x.

4. Proposed latent variable CRF models

This section introduces the proposed latent variable CRF models. In order to provide a clear explanation of the models, we briefly introduce the conventional CRF model, then present the proposed latent variable CRF models, and finally explain the main difference between these models. The first proposed model is a latent variable CRF-I, named LVCRF-I, which is a sentence level model and can automatically determine the best encoding schema for sequence labeling. The second proposed model is a latent variable CRF-II, named LVCRF-II, which is a word-level model that hybrids the path in the BIO encoding schema and the path in the BILOU encoding schema. This can enhance the prediction accuracy compared to the first proposed model because the best encoding schema for every word can be determined. The proposed models are described in details in the following.

4.1. Conventional CRF

The conditional random field (CRF) represents a popular model for the sequence labeling task. Compared with the other models, such as the hidden Markov model or maximum entropy model (MEM), CRF can easily incorporate flexible features and handle the label bias problem of the MEM model. The structure of the conventional CRF without using any encoding schema is presented in

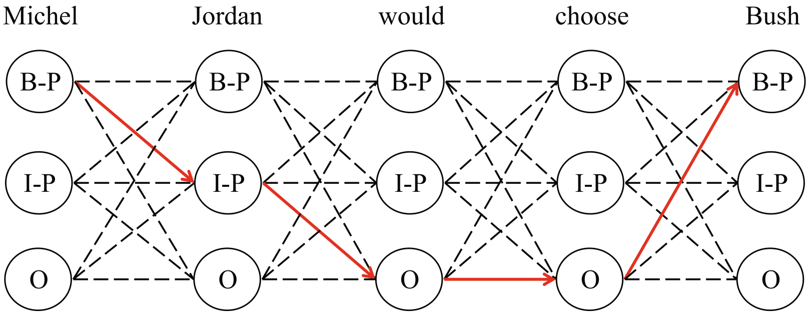


Fig. 2. The BIO encoding schema.

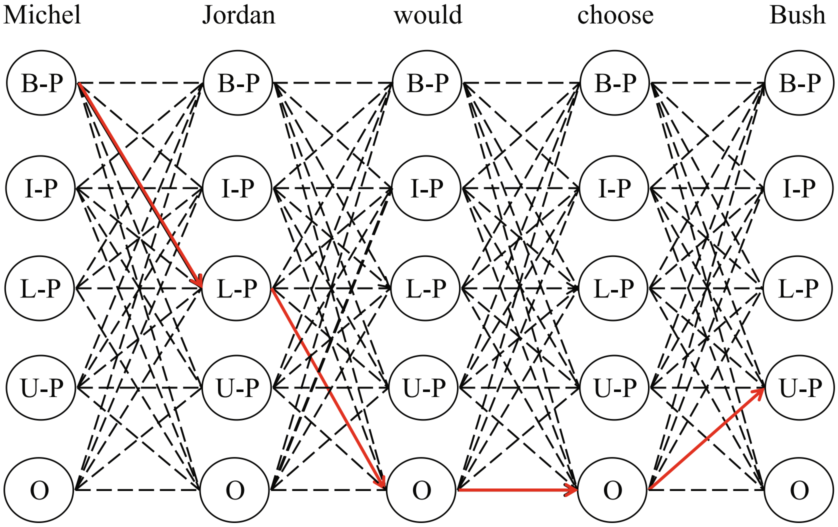


Fig. 3. The BILOU encoding schema.

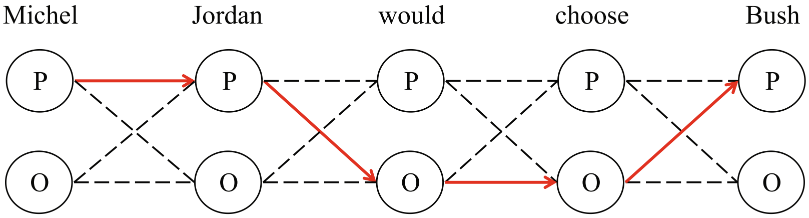


Fig. 4. Conventional CRF.

Fig. 4, where P node denotes the person name entity node and O node denotes the non-entity node. In Fig. 4, the dashed lines encode all the possible labeled paths of the given input sequences. Since the supervised training is utilized in the designed model, there is a labeled path (i.e., the red line) in the CRF model, which corresponds to a given label. During the training procedure, the model parameters are optimized to maximize the probability of the labeled path. The CRF model provides the conditional probability of a possible output sequence $s$for the input sequence x, which is given by:

$$p(s|x) = \frac{1}{Z(x)} exp\{W \cdot G(x,s)\}, \tag{2}$$

where $G(x,s)$ denotes the feature function, W denotes the weight vector, and $Z(x)$denotes the normalization factor. In order to find the best label sequence in CRF, let σ_j denote the best label sequence ends of the j-th input, (m,n,y) denote a label sequence that starts at the m-th position, end at n-th position, and is labeled as y. Then, $\sigma_j$can be recursively calculated as:

$$\sigma_j = max\Psi(j-1,j,y) + \sigma_{j-1}, \tag{3}$$

where $\Psi(j-1,j,y)$is the feature value defined over the label sequence $s = (j-l,j,y)$.

4.2. Latent Variable CRF-I

Compared with the conventional CRF, the proposed model incorporates hidden variables to explore more information from the input sequence. The structure of the proposed latent variable CRF is presented in Fig. 5.

As shown in Fig. 5, the proposed graph model consists of two parts. The upper part includes CRF with the BIO encoding schema,

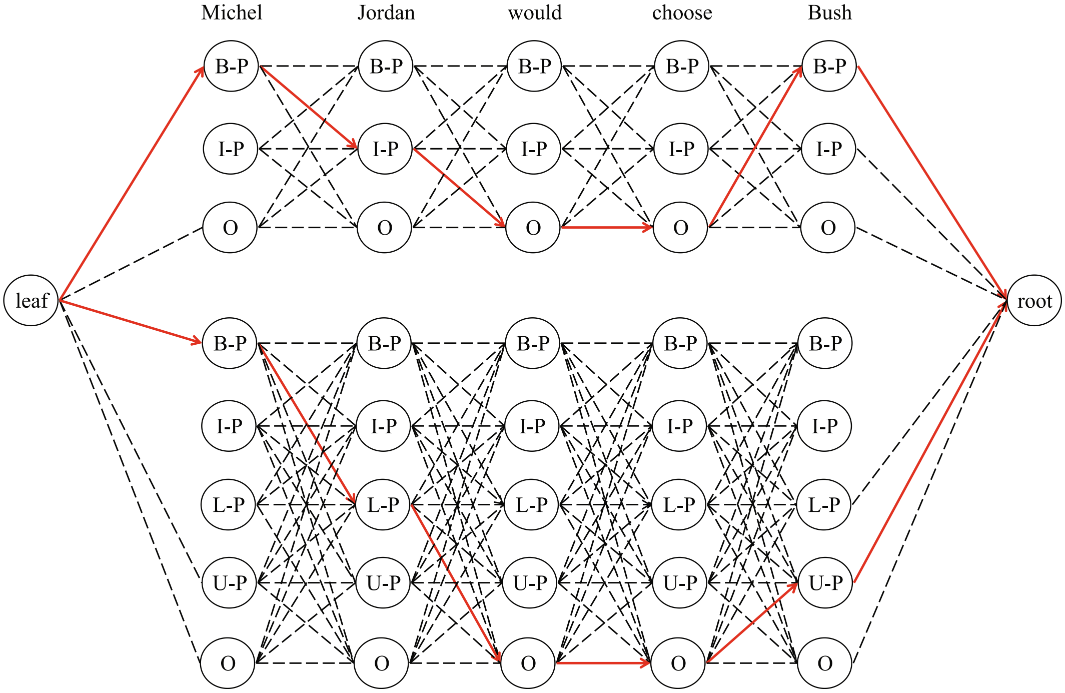


Fig. 5. The proposed latent variable CRF I (LVCRF-I).

which is introduced in Section 3.2. The connection relation is as follows. Node **B** can be connected to **I**, indicating that there is an entity that starts at the current position and continue to the next token, or to **O** and **B** nodes indicating that there is an entity of a unit length at the current position. Node **I** can be connected to **I**, meaning that there is an entity continuing to the next token, or to **O** and **B** nodes, meaning that there is an entity ending in that node. Node **O** can be connected to nodes **B** and **O**, suggesting that there is no entity at the current position, while it cannot be connected to node **I** because the begging of a segment has to be labeled with **B**. The bottom part in Fig. 5 corresponds to CRF with the BILOU encoding schema. The connection relation is as follows. Node **B** can be connected to nodes **I** and **L**, suggesting that there is an entity that starts at the current position, whereas the nodes cannot be connected to nodes **B**, **O**, and **U**, as a segment with a unit length should be labeled with **U**. Node **I** can be connected to nodes **I** and **L** because it denotes the inside of a segment, and thus, cannot be connected to nodes **B**, **U**, and **O** that denote the beginning of a new segment. Node **L** can be connected to nodes **B**, **U**, and **O**, meaning that the entity ends at the current position, but cannot be connected to nodes **L** and **I** because it denotes the ending of a segment. Nodes **U** can be connected to nodes **B**, **O**, and **U**, suggesting that there is an entity with a unit length at the current position, and cannot be connected to nodes **I** and **L** because there should be node **B** that denotes the beginning of a segment before them. Node **O** can be connected to nodes **B**, **O**, and **U**, suggesting that there is no entity at the current position, but cannot be connected to nodes **I** and **L** because there should be node **B** before them. The leaf node in the left part in Fig. 5 is denoted as the beginning of the sentence, and the root node in the right part is represented as the ending of the sentence.

As shown in Fig. 5, for a given input sentence, the proposed graph model provides two separate labeled paths, i.e., a path that corresponds to the BIO encoding schema and a path that corresponds to the BILOU encoding schema, respectively. It is worth mentioning that these two paths are the same; both path label “Michel Jordan” and “Bush” as name entity and “would choose” as a non-name entity. The only difference is that they use different encoding schemas. It should be noted that we do not tell the model whose encoding schema is better for sentences explicit in training phrase. We just label the sentence with both encoding schema and let the model learns by itself from the training data. In the decoding step, the model uses the Viterbi algorithm to choose one of the red paths with the highest feature scores as the final output label sequence. By using this framework, for a given input sentence, the model determines the encoding schema automatically, i.e., it uses either BIO or BILOU. This is the main difference compared to the conventional CRF model. In the conventional CRF models presented in Figs. 2–4, there is only one labeled path in the models that include one particular or none encoding schema. Thus, only limited information can be used in conventional CRF models. In real applications, different encoding schema can lead to different performance due to different models and different sequence-labeling methods. For instance, in the LVCRF-I model, the input sentence can be labeled by two encoding schemas simultaneously. During the learning phase, the model optimizes the parameters to maximize the probability of both encoding schemas. In the inference phase, the model outputs the encoding schema with the maximal probability. The model learns which encoding schema is better for a given input sentence by learning from the training data. The LVCRF-I represents a framework that can achieve better performance without using hand-crafted features. A similar framework can also be implemented using the other sequence labeling models, such as the HMM. Therefore, in this framework, the CRF models can be replaced with the HMM models. Also, this framework can be used in the neural-based sequence labeling models, such as the LSTM-CRF, by replacing the CRF layers in the neural CRF models with our framework (LVCRF-I).

4.3. Latent variable CRF-II

The latent variable CRF I can be considered as a sentence level model because it chooses the encoding schema for each sentence directly. Compared with the latent variable CRF I, the latent variable CRF II represents a word-level model because it chooses the encoding schema for each word automatically. The proposed latent variable CRF-II is presented in Fig. 6, where it can be seen that the proposed graph model consists of two parts.

The LVCRF-II has a similar structure with the LVCRF-I excepts that there are edges connecting the upper CRF and the bottom CRF. For example, the upper node **B** can be connected to bottom node **I**, meaning that there is an entity with BIO encoding schema that starts at the current position and continue to the next token with BILOU encoding schema, or to node **L** in the bottom CRF, denoting there is an entity of a unit length at the current position with BILOU encoding schema. The difference between LVCRF-I and LVCRF-II is that LVCRF-II allows the transform of the encoding schema given a certain input sentence. In LVCRF-I, two separate labeled paths will guide the model to learn in training phrase. In the decoding phrase, the model will use Viterbi algorithm (a dynamic programming algorithm) to choose the labled path with the highest score. If the model is correctly learned, then one and only one of the two labeled paths will be outputted. We force the model to choose a better encoding schema for the whole sentence. Thus, LVCRF-I learn the encoding schema in a sentence level. Compared with LVCRF-I, The LVCRF-II learns the encoding schema in a word level since there is a composite labeled path in LVCRF-II to guide the model learning. This labeled path labels each word with both BIO encoding schema and BILOU encoding schema, and allows transformation between each other. It can be seen that the two labeled paths in LVCRF-I is a subset of the labled paths in LVCRF-II. Still, only the path with highest feature scores are outputted in the decoding phrase. By doing so, more candidate labeled paths can be considered, and the model can choose the encoding schema for each word rather than for the whole sentence at once as LVCRF-I does.

Consider the example shown in 6. For a given input sentence, there are 2^n labeled paths, where n denotes the sentence length. All these paths are equal, i.e., they all label "Michel Jordan" and "Bush" as a name entity and "would choose" as a non-name entity. In the decoding step, the proposed model provides a subset of red lines, where lines are end-to-end connected to each other. As a result, each word can be labeled using different encoding schemas. For instance, 'Michel' can be labeled as nodes **B-P** using the BIO encoding schema, while 'Jordan' can be labeled as nodes **L-P** using the BILOU encoding schema. .

4.4. Training, inference and decoding

Following the CRF model, we adopt a log-linear approach as our objective function, which is expressed as:

$$L(w) = \sum_i log \sum_{y'} exp(w^T f(x_i, y')) - \sum_i w^T f(x_i, y) + \lambda w^T w, \quad (4)$$

where (x_i, y_i) refers to sentence x_i and the correct labeled path y_i, and the last term represents a $L2$regularization term with $\lambda$of 0.01. This objective function can be optimized by the standard gradient-based methods.

Specifically, for a given input sentence x, the probability of predicting a possible output sequence $y$is expressed as:

$$p(y|x) = \frac{exp(w^T f(x, y))}{\sum_{y'} exp(w^T f(x, y'))}, \quad (5)$$

where $f(x, y)$denotes the feature vector defined over the input–output pair (x, y), and the weight vector $w$provides the model parameters.

An inside-outside algorithm similar to the one presented in [25] is used in the inference process. The inference algorithm first calculates the inside score of each node from the leaf to the root node, and then the outside score from the root to the leaf node. The internal score is then calculated by summing up the features scores associated with the edge linking current node and its child nodes, while the internal score is calculated by the bottom-up (left-to-right) dynamic programming process. The path score is defined as a product of the inside score stored in the child node and the feature score defined over the edge connecting them. The computation of the outside score can be done in a similar manner from right to left. The inside (outside) score at a certain step can be calculated with a time complexity of $O(N^2)$, where N denotes the

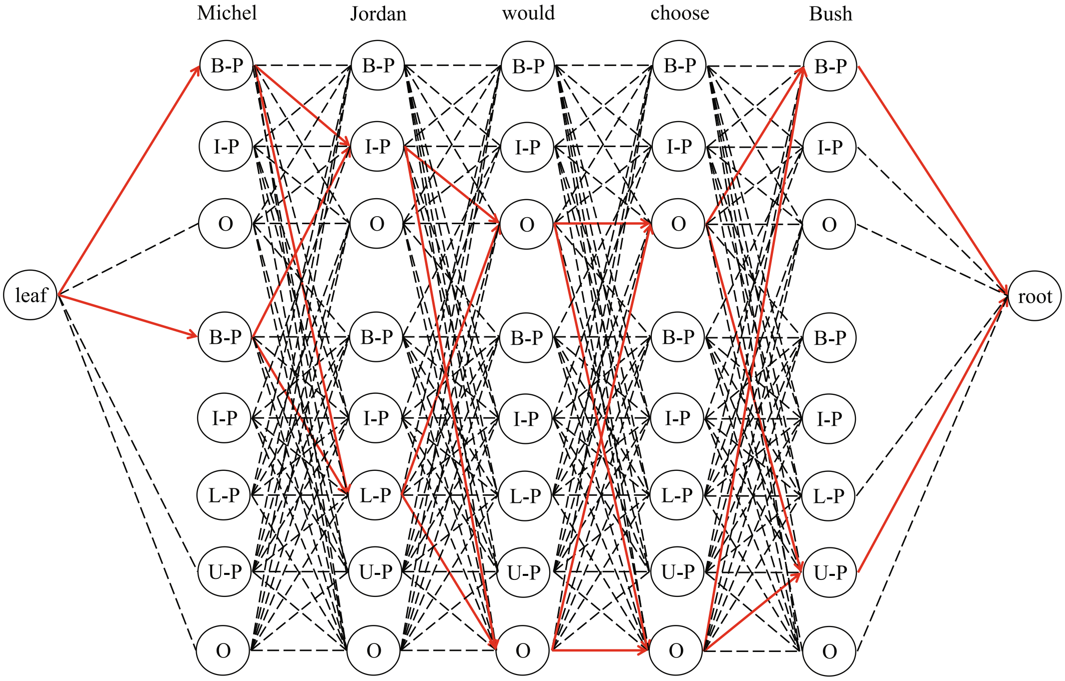


Fig. 6. The proposed latent variable CRF II (LVCRF-II).

number of entity types. This is because each node can be connected with at most $2N$ nodes (2 for two different encoding schemas), and there are $2N$ nodes at each time step($2N * 2N = O(N^2)$). Thus, for an input sentence with length T and N entity types, the time complexity of the proposed model is $O(TN^2)$, which is similar to that of the conventional CRF models. The Viterbi decoding algorithm (a dynamic programming algorithm) is utilized to obtain the output path with the highest probability. The training algorithm is similar to that utilized in conventional graphic model (forward–backward algorithm), which is shown as follows:

Algorithm 1 forward-backward algorithm

```
for each epoch do
    for each batch do
        (1) Forward pass to compute inside score $\alpha$
        (2) Backward pass to compute inside score $\beta$
        (3) Updating model parameters $\pi$ using $\alpha$ and $\beta$
    end for
end for
```

4.5. Features

We briefly introduce the CRF features to compute **G(x,s)** in Eq. 2. Specifically, we consider the following input features.

- **Word features:** Words that appear around the current position with a window size of three.
- **POS tag features (if available):** POS tags that appear around the current position with a window of size three.
- **Word *n*-gram features:** Word *n*-gram that contain the position, for n = 2, 3, 4.
- **POS *n*-gram features (if available):** POS tags that contain the current position, for n = 2, 3, 4.

All these features are used both in the conventional CRF based models and the proposed latent variable models for the purpose of comparisons. It is worth mentioning that the features used in this work are simple but still show good performance in terms of accuracy. However, the aim of this work is to evaluate the effectiveness of the proposed framework rather than feature engineering. Besides, the motivation of this paper is to present a framework without task-specific feature engineering.

5. Experimental evaluation

In this section, we evaluate our models on four natural language processing tasks, i.e., name entity recognition, chunking, reference parsing and POS tagging. We thoroughly compare the performance of the proposed latent variable CRF with the conventional CRF with BIO and BILOU encoding schemas. The developed models are released in Github (https://github.com/shaoyn0817/LVCRF-Model-Code). As mentioned before, our model can also be extended to the neural CRF based model. Thus, the neural CRF based model (i.e., LSTM-CRF) is also used in the comparison. The LSTM model utilized in our experiments is a conventional bi-directional LSTM model with a hidden size of 64. A task-specific pre-trained word embedding (64 dimension) is utilized as input of the neural based models. In the comparison, the CRF layer is replaced with our proposed LVCRF-I and LVCRF-II, obtaining the LSTM-LVCRF-I and LSTM-LVCRF-II, respectively. The CRF-BIO represents the conventional CRF with the BIO encoding schema, whereas the CRF-BILOU represents the conventional CRF with the BILOU encoding schema. The comparison of the LVCRF-I and LVCRF-II with the other models is as follows. In comparison, the models use the same feature as that described in Section 4.5.

5.1. Datasets

In the experiments, standard datasets were used to evaluate the performance of all the models. Table 1 lists the corpora statistics of the used datasets.

- **Conll2003:** We performed the experiments on the named entity recognition using the English data from a CoNLL 2003 shared task [17]. This dataset contained four different types of named entities, i.e., Person, Location, Organization, and Misc.
- **BC2GM:** We performed the experiments on the BioCreative II Gene Mention corpus that consisted of 20,000 sentences from the abstracts of biomedical publications, and it was annotated for the names of genes, proteins, and related entities using a single NE class.
- **JNLPBA:** We performed experiments on the JNLPBA corpus that consisted of 2,404 biomedical abstracts and was annotated for five entity types, i.e., CELL LINE, CELL TYPE, DNA, RNA, and PROTEIN. The corpus was derived from the GENIA corpus entity annotations for use in the shared task organized in conjunction with the BioNLP 2004 workshop.
- **CHEMDNER:** We performed experiments on the BioCreative IV Chemical and Drug NER 386 corpus [35] that consisted of 10,000 abstracts annotated for mentioning of chemical and drug 387 names using a single class.
- **Conll2000:** We performed experiments using the English data on a CoNLL 2000 shared task [23] for the chunking task. The Wall Street Journal Sections 15–18 from the Penn Treebank were used for training, and Section 20 was used for the test.
- **Cora:** We performed experiments on the reference parsing using the Cora dataset. Cora [36] contained 500 reference strings labeled by 13 fields, including the author, title, book title, journal, volume, pages, note, tech, date, editor, location, institution, and publisher.
- **PTB POS:** We performed experiments on the POS tagging using Penn TreeBank (PTB) POS tagging dataset. This dataset contains 30,000 sentences with 45 syntactic label.

5.2. Named entity recognition

Tables 2–5 compare the performance of different models, i.e., the NER task on the CoNLL2003, BC2GM, JNLPBA, and CHEMDNER datasets, where the best performance is marked with underline. As mentioned previously, the LVCRF-I can be viewed as a combination of the CRF-BIO and CRF-BILOU, which is why its performance was robust and outperformed the performances of both the CRF-BIO and the CRF-BILOU. This result proved that the LVCRF-I could automatically choose the best encoding schema for the input sentence. We also replaced the CRF layer in the LSTM-CRF model to obtain the LSTM-LVCRF-I and LSTM-LVCRF-II. It can be seen that the proposed LVCRF-I and LVCRF-II had better performance than the conventional CRF layer. The best performance was achieved by the LSTM-LVCRF-II. In this work, we propose a framework of the CRF that uses the encoding schema as a latent variable. The result showed that the proposed framework could easily outperform the conventional CRF model, as both an independent model and a layer in a neural-based model. As expected, the performance of the LVCRF-II was slightly better than that of the LVCRF-I. The CRF-BIO and CRF-BILOU exhibited poor performance, so is the performance of the CRF with different encoding schemas.

Table 1
Corpora Statistics of the used datasets.

Name	Task	# labels	# train	# dev	# test
CoNLL2003	NER	8	14,987	3,466	3,684
BC2GM	NER	3	12,500	2,500	5000
JNLPBA	NER	11	18,546	N/A	3,856
CHEMDNER	NER	3	2,916	2,907	2,478
CoNLL2000	Chunking	22	8,936	N/A	2,012
Cora	Ref parsing	13	500	N/A	N/A
PTB POS	Ref parsing	45	39831	1699	2415

Table 2
Results on the Conll2003 dataset.

NER task	Precision	Recall	F1
CRF-BIO	84.10	83.59	83.84
CRF-BILOU	83.82	84.36	84.09
LSTM-CRF	90.21	91.38	90.79
LVCRF-I	84.19	84.71	84.46
LVCRF-II	84.15	85.05	84.59
LSTM-LVCRF-I	90.11	92.15	91.12
LSTM-LVCRF-II	90.78	91.89	**91.33**

Table 3
Results on the BC2GM dataset.

NER task	Precision	Recall	F1
CRF-BIO	86.5	87.88	87.18
CRF-BILOU	86.88	88.05	87.46
LSTM-CRF	89.45	90.36	89.90
LVCRF-I	86.81	89.25	88.01
LVCRF-II	86.78	89.39	88.06
LSTM-LVCRF-I	90.65	91.12	90.88
LSTM-LVCRF-II	90.88	91.82	**91.34**

Table 4
Results on the JNLPBA dataset.

NER task	Precision	Recall	F1
CRF-BIO	70.23	71.18	70.70
CRF-BILOU	71.01	70.88	70.94
LSTM-CRF	73.34	70.89	72.09
LVCRF-I	71.12	71.01	71.06
LVCRF-II	71.22	71.20	71.20
LSTM-LVCRF-I	73.78	71.21	72.47
LSTM-LVCRF-II	73.26	71.76	**72.50**

Table 5
Results on the CHEMDNER dataset.

NER task	Precision	Recall	F1
CRF-BIO	82.21	83.92	83.05
CRF-BILOU	82.62	84.16	83.38
LSTM-CRF	84.79	86.21	85.49
LVCRF-I	82.59	84.38	83.47
LVCRF-II	83.71	84.35	84.02
LSTM-LVCRF-I	85.11	86.99	86.03
LSTM-LVCRF-II	85.02	87.35	**86.17**

Table 6
Results on the CoNLL2000 dataset.

Chunking task	Precision	Recall	F1
CRF-BIO	90.15	89.89	90.01
CRF-BILOU	90.05	89.88	89.96
LSTM-CRF	92.34	90.78	91.55
LVCRF-I	90.12	90.23	90.17
LVCRF-II	90.08	90.41	90.24
LSTM-LVCRF-I	92.72	91.43	92.07
LSTM-LVCRF-II	92.82	92.07	**92.44**

5.3. Chunking

Table 6 compares the performance of different models in the chunking task on the CoNLL2000 dataset [10]. As given in Table 6, the proposed models outperformed the baseline models, the CRF-BIO, the CRF-BILOU, and the LSTM-CRF. The LVCRF-I and LVCRF-II achieved the same performances in this task. The CRF with the BIO encoding schema performed better in the chunking task, while the CRF with the BILOU encoding schema was better in the named entity recognition. This was because none of the encoding schemas was the best for all the cases, so it was necessary to use different encoding schemas for different input sentences proposed in this work. On the CoNLL2000 dataset, the LSTM-LVCRF-II also achieved the best performance among all the models.

5.4. Reference parsing

The reference parsing provides more segment level information in comparison to the chunking and named entity recognition. Table 7 compares the performance of the methods in reference parsing using the Cora dataset [25]. For two compared baseline models, i.e., the CRF-BIO and the CRF-BILOU, the CRF-BILOU outperforms the CRF-BIO. This could be due to the fact that the CRF-BILOU could capture more segmental level information, i.e., boundary words, which was quite important in this task. The performance of the two proposed models was quite robust, and they both outperformed the CRF-BILOU model and the LSTM-CRF model.

5.5. POS tagging

POS tagging is the task of assigning each word with a syntactic tag given an input sentence. Compared with the above three tasks (i.e., reference parsing, chunking and NER), POS tagging has less segment level information. The performance of the two proposed models still outperformed the CRF models and the LSTM-CRF models, which can be observed in Table 8.

Table 7
Results on the Cora dataset.

Reference parsing task	Precision	Recall	F1
CRF-BIO	77.92	80.61	79.24
CRF-BILOU	78.35	81.21	79.75
LSTM-CRF	78.99	82.01	80.47
LVCRF-I	78.15	81.56	79.81
LVCRF-II	78.25	81.89	80.02
LSTM-LVCRF-I	79.38	81.78	80.56
LSTM-LVCRF-II	80.12	82.05	**81.07**

Table 8
Results on the Cora dataset.

Reference parsing task	Precision	Recall	F1
CRF-BIO	93.41	95.99	94.68
CRF-BILOU	93.45	95.27	94.35
LSTM-CRF	96.10	95.37	95.73
LVCRF-I	94.22	95.51	94.86
LVCRF-II	94.71	95.19	94.95
LSTM-LVCRF-I	95.92	96.26	96.08
LSTM-LVCRF-II	96.31	95.91	**96.11**

6. Conclusion

This paper studies the sequence labeling problem, which is often used as a pre-processing step in the natural language processing tasks, and it can help machines better understand the structure or components of a given text. We have carried out in-depth study on the performance of using different encoding schemas and propose two latent variable CRFs to improve the sequence labeling performance. The proposed LVCRF-I model can choose the best encoding schema for a given input sentence, while the proposed LVCRF-II model can choose the best encoding schema for every word in the input sequence. The effectiveness of the proposed models in several standard sequence labeling tasks on a few datasets was evaluated experimentally. Our future work will include the analysis of different encoding schemas, the development of new encoding schemas, and the design of an improved model by composing more encoding schemas as the latent variable.

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.

CRediT authorship contribution statement

Jerry Chun-Wei Lin: Writing - original draft, Writing - review & editing, Supervision. **Yinan Shao:** Writing - original draft. **Ji Zhang:** Validation. **Unil Yun:** Investigation.

References

[1] M. Mintz, R.S.S. Bills, D. Jurafsky, "Distant supervision for relation extraction without labeled data," in The Annual Meeting of the Association for, Comput. Ling. (2009) 1003–1011.
[2] P. Gupta, B. Andrassy, "Table filling multi-task recurrent neural network for joint entity and relation extraction, in: The International Conference on Computational Linguistics, 2016, pp. 2537–2547.
[3] M. W. C. S. Guo, E. Kiciman, To link or not to link? a study on end-to-end tweet entity linking, in: The Conference of the North American Chapter of the Association of Computational Linguistics, 2013, pp. 1020–1030.
[4] J. Lu, D. Venugopal, V. Gogate, V. Ng, "Joint inference for event coreference resolution,", in: The International Conference on Computational Linguistics, 2016, pp. 3264–3275.
[5] L. Ratinov, D. Roth, Design challenges and misconceptions in named entity recognition, in: The Conference on Computational Natural Language Learning, 2009, pp. 147–155.
[6] P.L.H. Dai, Y. Chang, R.T. Tsa, Enhancing of chemical compound and drug name recognition using representative tag scheme and fine-grained tokenization, J. Cheminform. 7 (S-1) (2015) 1–10.
[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] A.L. Berger, S.A.D. Pietra, V.J.D. Pietra, A maximum entropy approach to natural language processing, Comput. Ling. 22 (1) (1996) 39–71.
[11] J.D. Lafferty, A. Mccallum, F.C.N. Pereira, "Conditional random fields: probabilistic models for segmenting and labeling sequence data," in, in: The Eighteenth International Conference on Machine Learning, 2001, pp. 282–289.
[12] S. Sarawagi, W.W. Cohen, "Semi-markov conditional random fields for information extraction," in The Neural, Inform. Process. Syst. (2004) 1185–1192.
[13] Y.S.S. Fine, N. Tishby, The hierarchical hidden markov model: analysis and applications, Mach. Learn. 32 (1) (1998) 41–62.
[14] H.P. Zhang, Q. Liu, X.Q. Cheng, H. Zhang, H.K. Yu, "Chinese lexical analysis using hierarchical hidden markov model," in The Workshop on, Chinese Lang. Process. (2003) 63–70.
[15] D. Shen, J. Zhang, G. Zhou, J. Su, C.L. Tan, "Effective adaptation of a hidden markov model-based named entity recognizer for biomedical domain, The Nature Language Processing in Biomedicine, 2003, pp. 49–56.
[16] A. McCallum, D. Freitag, F.C.N. Pereira, Maximum entropy markov models for information extraction and segmentation, in: The International Conference on Machine Learning, 1999, pp. 591–598.
[17] D. Yu, L. Deng, A. Acero, Using continuous features in the maximum entropy model, pattern recognition letters, Pattern Recogn. Lett. 30 (14) (2009) 1295–1300.
[18] A. Ratnaparkhi, A maximum entropy model for part-of-speech tagging, in: The Conference on Empirical Methods in Natural Language Processing, 1996, pp. 133–142.
[19] D.S. Rosenberg, K. Dan, B. Taskar, Mixture-of-parents maximum entropy markov models, http://arxiv.org/abs/1206.5261, 2012.
[20] H. Tseng, P. Chang, G. Andrew, D. Jurafsky, C. Manning, "Sequential labeling with latent variables, The Workshop on Chinese Language Processing, 2015, pp. 168–171.
[21] H. Zhao, C.N. Huang, M. Li, T. Kudo, "An improved chinese word segmentation system with conditional random field," in The Workshop on, The Workshop on Chinese Language Processing, 2006, pp. 162–165.
[22] N.V. Cuong, W.S.L.N. Ye, L.C. Hai, Conditional random field with high-order dependencies for sequence labeling and segmentation, J. Mach. Learn. Res. 15 (1) (2014) 981–1009.
[23] 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, Comput. Ling. (2006) 465–472.
[24] V.C. Nguyen, W.S.L.N. Ye, L.C. Hai, Semi-markov conditional random field with high-order feature, 2011, pp. 1–4.
[25] 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, Comput. Ling.: Human Language Technol. (2016) 714–719.
[26] Z. Huang, W. Xu, K. Yu, Bidirectional lstm-crf models for sequence tagging, http://arxiv.org/abs/1508.01991s, 2015.
[27] Y. Liu, W. Che, J. Guo, Q. Bin, T. Liu, Exploring segment representations for neural segmentation models, in: The International Joint Conference on Artificial Intelligence, 2016, pp. 2880–288.
[28] X. Ma, E. Hovy, "End-to-end sequence labeling via bi-directional lstm-cnns-crf," in The Annual Meeting of the Association for, Comput. Ling. (2016) 1064–1074.
[29] M. Rei, G.K.O. Crichton, S. Pyysalo, Attending to characters in neural sequence labeling models, http://arXiv:1611.04361, 2016.
[30] X. Sun, X. Nan, "Chinese base phrases chunking based on latent semi-crf mode," in, in: The International Conference on Natural Language Processing and Knowledge Engineering, 2010, pp. 1–7.
[31] S. Petrov, K. Dan, Sparse multi-scale grammars for discriminative latent variable parsing, in: The Conference on Empirical Methods in Natural Language Processing, 2008, pp. 867–876.
[32] X. Sun, D. Huang, F. Ren, Detecting new words from chinese text using latent semi-crf models, IEICE Trans. Inform. Syst. 93 (6) (2010) 1386–1393.
[33] X. Sun, J. Tsujii, "Sequential labeling with latent variables," in The European Chapter of the Association for, Comput. Ling. (2009) 772–780.
[34] X. Sun, S. Sun, M. Yin, H. Yang, Hybrid neural conditional random fields for multi-view sequence labeling, Knowl.-Based Syst. 189 (2020) 105151.
[35] M. Krallinger, F. Leitner, O. Rabal, M. Vazquez, J. Oyarzabal, A. Valencia, Chemdner: The drugs and chemical names extraction challenge, J. Cheminform. 7 (S1) (2015) 1–12.
[36] L.E. Baum, G.R. Sell, Growth transformations for functions on manifolds, Pac. J. Math. 27 (2) (1968) 211–227.

Jerry Chun-Wei Lin received his Ph.D. from the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan in 2010. He is currently a full Professor with the Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway. He has published more than 300 research articles in refereed journals (IEEE TKDE, IEEE TCYB, ACM TKDD, ACM TDS, ACM TMIS) and international conferences (IEEE ICDE, IEEE ICDM, PKDD, PAKDD). His research interests include data mining, soft computing, artificial intelligence and machine learning, and privacy preserving and security technologies. He is also the project co-leader of well-known SPMF: An Open-Source Data Mining Library, which is a toolkit offering multiple types of data mining algorithms. He also serves as the Editor-in- Chief of the International Journal of Data Science and Pattern Recognition. He is the IET Fellow, senior member for both IEEE and ACM.

Yinan Shao received his Master degree from the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), Shenzhen, China. He is currently working as an Algorithm Engineer in Alibaba Inc., Hangzhou, China. His research interests include pattern mining, machine learning and deep learning, and natural language processing.

Professor Ji Zhang is currently an Associate Professor in computer science at the University of Southern Queensland (USQ), Australia. He is an IEEE senior member, ACM member, Australian Endeavour Fellow, Queensland Fellow (Australia) and Izaak Walton Killam Scholar (Canada). His research interests are Big data analytics, knowledge discovery and data mining (KDD), information privacy and security. He was a Postdoctoral Research Fellow in CSIRO ICT Center at Hobart, Australia from 2008–2009. He received his degree of Ph.D. from the Faculty of Computer Science at Dalhousie University, Canada in 2008. He has published over 150 papers in major peer-reviewed international journals and conferences.

Unil Yun received the M.S. degree in computer science and engineering from Korea University, Seoul, South Korea in 1997, and the Ph.D. degree in computer science from Texas A&M University, College Station, TX, USA, in 2005. He was with the Multimedia Laboratory, Korea Telecom, from 1997 to 2002. After receiving the Ph.D. degree, he was a Postdoctoral Associate for almost one year in the Computer Sciece Department, Texas A&M University. After that, he was a Senior Researcher with the Electronics and Telecommunications Research Institute. In March 2007, he jointed the School of Electronic and Computer Engineering, Chungbuk National University, South Korea. Since August 2013, he has been with the Department of Computer Engineering, Sejong University, Seoul, South Korea. His research includes data mining, information retrieval, database systems, artificial intelligence, and digital libraries.